

TRABAJO DE FINAL DE GRADO

GRADO EN INGENIERÍA INFORMÁTICA

ESPECIALIDAD EN INGENIERÍA DEL SOFTWARE

UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

Facultat d'Informàtica de Barcelona



# IMPLANTACIÓN DE UNA METODOLOGÍA PARA LA GESTIÓN DE PROYECTOS SOFTWARE

**Autor:** Jordi Fernández Cuevas

**Director:** Luis Mera Castellón

**Ponente:** Enric Mayol Sarroca

Trabajo desarrollado en colaboración de Develoop Software



## **RESUMEN**

El rol de un gestor de proyectos es velar por el buen desenlace de todo proyecto software, así como el cumplimiento de las fechas estipuladas con el cliente, y las funcionalidades requeridas.

Para ello el seguimiento de una metodología es necesario, ya que la buena definición y adaptación de esta en el marco de la empresa puede ayudar a que el proyecto sea un éxito, o por lo menos minimizar el impacto de posibles problemas que puedan surgir, así como de la improvisación a lo largo de un proyecto software.

## **RESUM**

El rol del gestor de projectes es velar pel bon desenvolupament del projecte software, així com del compliment de les dates concretades amb el client, i les funcionalitats requerides.

Per fer-ho possible, emprar una metodologia és necessari ja que una bona definició i adaptació d'aquesta en el marc de l'empresa pot ajudar a que el projecte sigui un èxit, o almenys a minimitzar l'impacte de possibles problemes que puguin sorgir, així com de la improvitació durant el projecte software.

## **ABSTRACT**

The aim of a project manager is to ensure the success of the project, as well as to satisfy the different deadlines accorded with the client and the project functionalities.

In order to make it possible, using a well defined and adapted methodology to company's environment is really important. It would help in the ideal outcome of the project, dealing with the different problems and improvisation during the software development.

## ÍNDICE

1. CONTEXTO.....	7
1.1. INTRODUCCIÓN.....	7
1.2. DESCRIPCIÓN DEL PROYECTO .....	7
1.3. PROBLEMÁTICA.....	9
1.4. STAKEHOLDERS (ACTORES IMPLICADOS).....	12
2. JUSTIFICACIÓN.....	14
2.1. APROVECHAR O DISEÑAR UNA SOLUCIÓN .....	14
3. ALCANCE.....	15
3.1. SUBOBJETIVOS .....	16
3.2. RIESGOS Y OBSTÁCULOS.....	16
4. METODOLOGIA Y RIGOR .....	17
4.1. METODOLOGIA .....	17
4.2. HERRAMIENTAS DE SEGUIMIENTO .....	17
5. DESCRIPCIÓN DE LAS TAREAS .....	19
5.1. GESTIÓN DE PROYECTO.....	19
5.2. ANÁLISIS .....	19
5.3. IMPLEMENTACIÓN.....	20
5.4. FASE FINAL .....	21
6 PLANIFICACIÓN .....	22
6.1. ESTIMACIONES.....	22
6.2. DIAGRAMA DE GANTT .....	23
7. GESTIÓN DE LOS RIESGOS .....	24
7.1. PLANES ALTERNATIVOS.....	24
8. PRESUPUESTO.....	26
8.1. IDENTIFICACIÓN DE COSTES.....	26
8.2. CONTINGENCIAS .....	28
8.3. CONTROL DE GESTIÓN .....	29
9. INFORME DE SOSTENIBILIDAD .....	30
9.1. DIMENSIÓN ECONÓMICA.....	30
9.2. DIMENSIÓN AMBIENTAL .....	31
9.3. DIMENSIÓN SOCIAL .....	31
10. ESTADO DEL ARTE.....	32
10.1. METODOLOGIA TRADICIONAL O EN CASCADA.....	32
10.2. METODOLOGIA ÁGIL .....	33
10.3. SCRUM .....	34

10.4. PRINCE2 .....	36
10.5. PMBOK.....	41
10.6. AGILE UNIFIED PROCES (AUP) .....	44
10.7. KANBAN.....	47
11. EMPRESA .....	50
11.1. EQUIPO DEVELOOP .....	50
11.2. HERRAMIENTAS PARA LA GESTIÓN.....	51
11.3. DESARROLLO DEL PROYECTO.....	52
11.4. ODOO PROYECTOS .....	56
12. DECISIÓN FINAL .....	60
13. IMPLEMENTACIÓN DE LA METODOLOGIA .....	63
13.1. CONCEPTOS PREVIOS .....	63
13.2. CONTACTO COMERCIAL.....	65
13.3. PRESUPUESTO.....	65
13.4. CONTRATO.....	66
13.5. ANALISIS FUNCIONAL Y DISEÑOS .....	66
13.6. DESARROLLO DEL PROYECTO.....	66
13.7. ENTREGA .....	72
13.8. MANTENIMIENTO.....	73
14. CONCLUSIÓN.....	74
14.1. DESVIACIONES .....	74
14.2. CONSIDERACIONES FINALES .....	76
14.3. REFLEXIONES.....	78
15. BIBLIOGRAFÍA.....	79
16. GLOSARIO .....	82

## ÍNDICE DE ILUSTRACIONES

Ilustración 1. Etapas en metodología en cascada. ....	32
Ilustración 2. Flujo en Scrum. ....	35
Ilustración 3. Proceso en PRINCE2. ....	39
Ilustración 4. Procesos PMBOK. ....	42
Ilustración 5. Fases en AUP. ....	45
Ilustración 6. Vista Kanban. ....	47
Ilustración 7. Vista Kanban en Odoo. ....	56
Ilustración 8. Ficha de tarea. ....	57
Ilustración 9. Vista Gantt en Odoo. ....	58
Ilustración 10. Triángulo de hierro. ....	64
Ilustración 11. Organización proyecto. ....	68
Ilustración 12. Ejemplo Burndown Chart. ....	71
Ilustración 13. Ejemplo Velocity Chart. ....	72
Ilustración 14. BPMN nueva metodología. ....	73

## ÍNDICE DE TABLAS

Tabla 1. Estimación de las tareas. Elaboración propia.....	22
Tabla 2. Diagrama de Gantt del proyecto. Elaboración propia. ....	23
Tabla 3. Costes humanos. Elaboración propia.....	27
Tabla 4. Costes de hardware. Elaboración propia.....	27
Tabla 5. Costes de software. Elaboración propia. ....	28
Tabla 6. Costes genéricos. Elaboración propia. ....	28
Tabla 7. Costes finales aplicando contingencias. Elaboración propia. ....	29
Tabla 8. Comparativa de metodologías. Elaboración propia. ....	61
Tabla 9. Diagrama de Gantt real. Elaboración propia. ....	75

# 1. CONTEXTO

## 1.1. INTRODUCCIÓN

Este Trabajo de Fin de Grado: “Implantación de una metodología para la gestión de proyectos software”; pertenece a los estudios del Grado en Ingeniería Informática (GEI) que se imparte en la Facultad de Informática de Barcelona (FIB) en la Universidad Politécnica de Barcelona (UPC), dentro de la especialidad de Ingeniería del Software.

Se trata de un trabajo de modalidad B realizado en la empresa Develoop Software, la cual se encarga del desarrollo de todo tipo de software a medida, según los requerimientos y especificaciones del cliente.

## 1.2. DESCRIPCIÓN DEL PROYECTO

Antes que todo y habiendo analizado el título hay una serie de palabras que nos ayudan a contextualizar el alcance de este TFG. Aparecen las palabras: “metodología”, “gestión de proyectos” y “software”.

Para ir entrando en materia se definirán estas *keywords* de una manera entendible y que no conlleve a error.

### 1.2.1. SOFTWARE

Empezaremos primeramente por analizar la palabra “software”. Es una palabra que, por el auge de la tecnología, la escuchamos cada vez más frecuentemente. ¿Pero que es realmente “software”?

*What/Is* lo define como:

“El software es un conjunto de instrucciones, datos o programas utilizados para operar ordenadores y ejecutar tareas específicas. A diferencia del *hardware*, que describe los aspectos físicos de un ordenador, el software es un término genérico utilizado para referirse a aplicaciones, scripts y programas que se ejecutan en un dispositivo.” [6]

El software ayuda al usuario a realizar las tareas de una manera mucho más fácil, en la que el principal objetivo es automatizar el máximo número de procesos a realizar, para que de esta manera se reduzca el error humano.

Cualquier software de calidad debería cumplir las características marcadas en la Norma ISO/IEC 25000, una actualización de la ISO/IEC 9126:

- Adecuación funcional
- Seguridad
- Compatibilidad
- Fiabilidad
- Usabilidad
- Eficiencia de desempeño
- Mantenibilidad
- Portabilidad [1]

### **1.2.2. GESTIÓN DE PROYECTOS**

Otra de las palabras claves importantes de entender del título es la “gestión de proyectos”, que *WhatIs* lo define como:

“La gestión de proyectos es la disciplina de utilizar, principios, procedimientos y políticas establecidas para guiar con éxito un proyecto desde la concepción hasta su finalización.” [6]

Todo proyecto debe estar dirigido por una persona que se encarga de aplicar los conocimientos, habilidades, herramientas y técnicas para cumplir con el requisito de este.

La manera en la que un proyecto se gestiona, no es homogénea para todos los proyectos, cada proyecto tiene sus propias particularidades. [5]

En este TFG la gestión de un proyecto va a abarcar desde el momento en el que el cliente contacta con la empresa en búsqueda de una solución, pasando por la llamada comercial, realización del presupuesto, análisis funcional, diseño e implementación del software, entrega del software al cliente hasta el servicio post-venta y/o posibles incidencias.



### **1.2.3. METODOLOGÍA**

El término metodología es muy común pero no viene de más recordar su significado que según la RAE es el siguiente:

“Conjunto de métodos que se siguen en una investigación científica o en una exposición doctrinal”

Hace mención al “método” que es:

“Modo de obrar o proceder, hábito o costumbre que cada uno tiene y observa.”

[4]

Por lo que una metodología para la gestión de proyectos software sería una hoja de ruta a seguir desde el inicio de un proyecto hasta que este se termina.

### **1.2.4. CONCLUSIÓN**

Tomando en cuenta todo lo comentado en los puntos anteriores, el principal propósito de este TFG, como bien indica el título, es implantar una metodología en la gestión de los proyectos, en la que se definan los pasos a seguir, las etapas en las que se dividen y todos aquellos puntos clave en el seguimiento de un proyecto software. Como así también del uso de las herramientas de las cuales se disponen para realizar dicho seguimiento.

## **1.3. PROBLEMÁTICA**

Develoop Software es una empresa que se lleva dedicando al sector de los servicios desde el 1997, desde entonces, como cualquier empresa del ámbito de las tecnologías de la información ha tenido que ir evolucionando debido a las necesidades del sector y los grandes avances tecnológicos. [2]

Actualmente, en Develoop nos encontramos en un período de actualización y mejora a nivel procedimental. Es por eso que decidimos que la estandarización de una metodología para la gestión de los proyectos nos ayudaría a crecer como empresa y, nos facilitaría el trabajo a todos los integrantes reduciendo así

la improvisación en las tareas a realizar, siguiendo los pasos establecidos en la metodología.

La actual metodología en cada uno de los proyectos, varía dependiendo del tipo de proyecto y/o cliente. Pero a grandes rasgos se realizan los siguientes pasos, característicos de una metodología tradicional (Posteriormente los acabaremos de detallar.):

- Se contacta con el cliente para tener una idea del proyecto.
- Se realiza un presupuesto, con las estimaciones de horas aportadas por el director técnico y posterior revisión del Project Manager.
- Se realiza el análisis funcional y el diseño.
- Se planifican las tareas en el tiempo. Si se requiere se detectan entregas intermedias.
- Se recoge *feedback* del cliente y se realizan las correcciones necesarias.
- Se realiza el QA. Se corrige lo detectado.
- Finalmente se pone en producción, previamente aprobado por el cliente.

Al no tener concretada una metodología, en algunos casos, los clientes esperan una manera de trabajar, pero por parte de Develoop se tenía prevista otra y esto desemboca en malestar por parte del cliente y retraso en la entrega del proyecto.

Sin embargo, tener documentada y estandarizada una metodología, con la que se va a trabajar en la mayor parte de los proyectos, habrá casos concretos los cuales no se va a poder aplicar, y conocida tanto por todos los miembros del equipo, como por el cliente, puede traer múltiples beneficios a todos los integrantes del proyecto.

Es por eso, que en este punto vamos a comentar algunos de los problemas detectados en los proyectos, para poderlos tener en cuenta en la futura elección de la metodología, de tal manera que esta nos ayude a paliar con estos problemas y ofrezca soluciones robustas que nos ayuden a reducir el impacto de estos problemas.

- Horas. El principal problema de los proyectos actuales son las horas reales que se dedican al proyecto, respecto a las que se habían estipulado desde un principio, sobre las cuales se había realizado el presupuesto y la planificación. Es uno de los temas que más preocupan ya que si se dedican más horas de las esperadas en un proyecto, significa que la viabilidad económica de proyectos futuros se pueden ver comprometida y por ende la de la empresa.
- Retrasos. Ligado al punto anterior, si se dedican más horas de las esperadas, sobre las cuales se monta la planificación del proyecto, puede desembocar en retrasos en la entrega. En el contrato del proyecto, se define un *deadline* para la entrega del proyecto, y a veces surgen retrasos que no hacen posible la entrega del proyecto en la fecha estipulada.
- Aprobación de nuevas funcionalidades. Actualmente, la mayoría de nuevas funcionalidades que no representan un gran impacto en el devenir del proyecto, se suelen aceptar, y es Develoop quien corre con los gastos extras, por pocos que sean. Creemos que este es uno de los puntos mejorables, y buscar la manera en que estos pequeños cambios no tengan un gran impacto en el proyecto.

Como podemos apreciar, todos estos problemas tienen una relación muy estrecha en su origen: la gestión del proyecto. Esto no implica, que la gestión actual de los proyectos sea desastrosa, sino que únicamente creemos que es mejorable. Y uno de los principales objetivos del proyecto, es ayudar a resolver estos problemas y estandarizar algunos procesos, que podrían ser cruciales en la gestión de los proyectos los cuales llevamos a cabo.

Hay que tener en cuenta, respecto al tercer problema comentado, que Develoop Software, como ya habremos comentado, es una empresa que pertenece al sector de los servicios, y con eso nos referimos que una de las principales satisfacciones para la empresa es el éxito del proyecto cara al cliente. Es por eso, que a veces, con tal de que el proyecto sea de su agrado, se aceptan algunas modificaciones no previstas, siempre y cuando aporten al proyecto.

Además, Develoop es una empresa que cree enormemente en la fidelización de los clientes y piensa que si el proyecto ha sido un éxito para el cliente, este puede volver a contratar los servicios de la empresa en futuros proyectos y contar con el mejor *marketing*: el boca a boca. Es esta la razón, por lo que a veces se sacrifican estas horas de más en tareas no contempladas.

## **1.4. STAKEHOLDERS (ACTORES IMPLICADOS)**

En este punto se van a listar los principales actores implicados, interesados y/o participantes en la implantación de esta metodología:

### **1.4.1. PROJECT MANAGER**

El Project Manager es el responsable del proyecto de software al cual está asignado, este se encarga de:

- Definir y presentar el proyecto.
- Planificar y definir fechas, plazos, recursos y costes.
- Establecer los objetivos.
- Implementar soluciones o cambios. [3]

Es por eso que las personas que desempeñan este rol, son las principales implicadas en este proyecto ya que está directamente relacionado con el trabajo que realizan dentro de un proyecto.

### **1.4.2. EQUIPO DE DESARROLLO**

El equipo de desarrollo es aquel encargado del desarrollo de un proyecto software: maquetar, diseñar e implementar la solución.

Este equipo es uno de los principales *stakeholders*, al estar vinculado directamente con el desarrollo del proyecto. Al fin y al cabo, va acabando desarrollando el producto pero siguiendo otra manera de proceder.

#### **1.4.3. CLIENTE**

La estandarización de una metodología de trabajo, no únicamente beneficia a la empresa en la cual es implantada, sino al cliente al cual se le realiza el proyecto, ya que conoce perfectamente la metodología y la manera de trabajar del equipo en el proyecto. Por lo que le interesa que, a parte de una buena coordinación entre el equipo, exista una manera de proceder que beneficie la organización y la puesta en marcha del proyecto. E incluso, la manera en la que se trabaja puede ser uno de los puntos determinantes para escoger a Develoop entre otras empresas.

#### **1.4.4. DEVELOOP SOFTWARE**

Todo aquello que implique una mejora en cualquier aspecto de una empresa, la que más se beneficia de ello es la propia empresa.

Esta implantación y documentación de la metodología enriquece a la empresa y la hace crecer en todos los sentidos. Tener una guía a seguir durante una gestión de un proyecto, unos pasos a seguir, puede ayudar a optimizar los tiempos, a planificar mucho mejor un proyecto y por lo tanto un mayor rendimiento.

## **2. JUSTIFICACIÓN**

### **2.1. APROVECHAR O DISEÑAR UNA SOLUCIÓN**

La creación de una nueva metodología a implementar en los proyectos de una empresa implicaría tener en cuenta todos y cada uno de los procesos que componen la gestión de un proyecto. Al fin y al cabo se daría con la solución idónea para la empresa, una solución poco madura y nada testeada, que se debería ir perfeccionando a medida que se vayan encontrando algunas lagunas e imperfecciones que no concuerdan con la manera de proceder de la empresa, o que, por ciertas razones, no se adaptarían a las necesidades.

Buscar una metodología que se adecue a la perfección a los intereses de una empresa es complicado, puede haber algunos aspectos que no concuerden del todo con la filosofía de la empresa. Pero una metodología es útil para una empresa cuando esta la beneficia, sea una metodología tradicional, sea una metodología ágil, de las cuales hablaremos más adelante, o sea una metodología inventada.

Es por eso que la mejor solución para este caso es aprovechar una solución ya existente la cual se haya probado en distintos proyectos y tengamos ya sus debilidades y fortalezas bien detectadas, y poder adaptar dicha metodología a nuestra manera de trabajar y cada una de nuestras necesidades.

Más adelante, analizaremos cada una de las metodologías que nos podrían interesar y acabaremos concretando con cuál de ellas proceder.

Como bien hemos dicho la metodología que seleccionemos la utilizaremos a método de plantilla, eso quiere decir que no tomaremos la metodología al pie de la letra sino que la adaptaremos a nuestro equipo y a las herramientas que debemos usar.

Un punto importante en este TFG es el detalle del uso de una nueva herramienta incorporada en la empresa la cual deberemos adaptar a la metodología que escojamos, por lo que el uso de esta deberá estar bien definido.

### 3. ALCANCE

En este punto vamos a decidir el alcance de este trabajo, es decir, lo que queda dentro y lo que queda fuera del proyecto.

Este proyecto cubrirá:

- Un estado del arte para recabar información de todas las metodologías existentes con las que podríamos llegar a trabajar.
- Un análisis de la manera en la que se trabaja actualmente, y las herramientas usadas.
- La implantación de la metodología en todo el proceso de la gestión de un proyecto: desde que llega un *lead* hasta el servicio post-venta.

Cabe resaltar que este TFG únicamente abarcará hasta la documentación de la metodología, es decir, la definición de todos los pasos a seguir en la gestión de un proyecto. La verificación de la metodología y el uso de esta, quedan fuera del alcance del trabajo.

A nivel de competencias técnicas, el TFG abarcará las siguientes:

- CES 1.2. Dar solución a los problemas de integración en función de las estrategias, de los estándares y de las tecnologías disponibles.
- CES 1.3. Identificar, evaluar y gestionar los riesgos potenciales asociados a la construcción del software que se puedan presentar.
- CES 2.1. Definir y gestionar los requisitos de un sistema *software*.
- CES 2.2. Diseñar soluciones apropiadas en uno o más dominios de aplicación, utilizando métodos de ingeniería del *software* que integren aspectos éticos, sociales y económicos.
- CES 1.9. Demostrar comprensión en la gestión y gobierno de los sistemas *software*.

### 3.1. SUBOBJETIVOS

Dentro de los puntos comentados con anterioridad podemos distinguir distintos subobjetivos en este trabajo:

- Conocer las diferencias entre las metodologías tradicionales y las metodologías ágiles. Así como ejemplificar metodologías de los ambos tipos.
- Estudiar qué metodología se adapta más a la manera de proceder de todo el equipo.
- Conocer la nueva herramienta incorporada en Develoop Software y sus beneficios a la hora de usarla en la gestión de los proyectos.
- Diferenciar cada una de las fases en la gestión de un proyecto.

### 3.2. RIESGOS Y OBSTÁCULOS

Es conocido que en cualquier tipo de proyecto existen una serie de obstáculos y riesgos que tienen que ser conocidos y tener presentes en todo momento. En este trabajo se han identificado los siguientes:

- Tiempo. Para este TFG hay una fecha final de entrega en la que tiene que estar completamente acabado, y aunque parece que haya tiempo suficiente no debemos confiarnos.
- Mala planificación del proyecto. Va muy ligado al punto anterior, ya que, si se planifica de manera errónea el trabajo, y no se cumplen los tiempos estimados para cada tarea, el proyecto se puede llegar a demorar y no estar listo cuando debería estar.
- Inexperiencia en el uso de la nueva herramienta para la gestión de proyectos. En este TFG se tiene previsto detallar el uso de la nueva herramienta incorporada en la empresa para la gestión de los proyectos. Al no tener experiencia en el uso de esta, puede conllevar emplear un tiempo no previsto en la familiarización con el entorno.



## 4. METODOLOGIA Y RIGOR

### 4.1. METODOLOGIA

La metodología a seguir en este TFG va ser una metodología en cascada la cual cada uno de los procesos a realizar es dependiente del anterior, por lo que las fases fijadas deberán ser completadas para avanzar a la siguiente. Las fases contempladas son las siguientes:

- Requisitos. En la que se va a analizar la manera de trabajar en Develoop, y se va a decidir la metodología la cual vamos a tomar como plantilla.
- Diseño. Se va a trabajar sobre la metodología escogida y se van a definir en detalle cada uno de los pasos a seguir.
- Implementación. En esta fase, revisaremos como aplicarla a las herramientas que usamos para la gestión de los proyectos.
- Verificación. Se comprobaran que todos los procesos son los adecuados y encajan a la perfección con lo esperado.
- Mantenimiento. En esta fase se irá depurando la metodología y ajustando cada uno de los procesos definidos a medida que se vaya siguiendo en los distintos procesos. Como hemos comentado anteriormente, este punto queda fuera del alcance del proyecto.

Aunque se va a seguir uno de los hábitos típicos de las metodologías ágiles, en el que se van a realizar reuniones periódicas de seguimiento junto al director del TFG, que nos permitan tener controlados los tiempos especificados.

### 4.2. HERRAMIENTAS DE SEGUIMIENTO

Para realizar un buen seguimiento y gestión del TFG, se usarán las siguientes herramientas que ayudarán a la contabilización de las horas y tener presentes las tareas a realizar:

Trello. Es un software utilizado para la gestión de los proyectos en los cuales se crean tarjetas, que equivalen a tareas, y cada una puede estar asignada a

un estado. Estas se organizan en una vista *Kanban*, la cual ayuda a tener una noción general del estado de cada una de las tareas.

Manda Horas: Herramienta usada a nivel interno en la empresa para la contabilización de las horas dedicadas a cada proyecto.

Proyectos de Odoo. Nueva herramienta incorporada en la empresa, la cual nos va a ayudar junto a Trello a la gestión de los proyectos. En las cuales, a cada una de las tareas creadas, se puede contabilizar las horas dedicadas.

## 5. DESCRIPCIÓN DE LAS TAREAS

En este punto se va a proceder a describir cada una de las tareas que conforman este TFG, así como la estimación de horas para cada una de ellas. Se van a dividir las tareas en grandes grupos y se detallaran cada uno de ellos, en caso oportuno se concretaran las subtareas individualmente.

### 5.1. GESTIÓN DE PROYECTO

Este grupo de tareas está conformado en su mayor parte por las tareas a realizar en el curso de GEP. Donde el principal objetivo es realizar una planificación del trabajo así como la contextualización y el alcance de este. Los contenidos de esta parte en específico son:

- Contextualización y alcance.
- Planificación temporal.
- Presupuesto y sostenibilidad.
- Entrega final.

A parte de comprender los puntos de GEP, para tener un mayor control del *planning* establecido, se realizarán reuniones periódicas con el director del TFG para tener un control de todo lo realizado y lo pendiente de realizar, así como el detalle de posibles correcciones a realizar.

En la gestión del proyecto se requerirán los siguientes recursos:

- **Recursos materiales:** Ordenador.
- **Recursos humanos:** Gestor de proyectos.

### 5.2. ANÁLISIS

La realización de un buen análisis de las soluciones existentes, así como de la manera de trabajar en la empresa es uno de los puntos clave del proyecto, ya que es en base a esta investigación donde se va a decantar por una metodología u otra. Dentro de este grupo de tareas podemos distinguir entre:

- Estado del arte. Análisis a fondo de las metodologías ágiles y tradicionales que más pueden convenir a estudiar, así como identificación de los pros y los contras de cada una de las metodologías.
- Manera de trabajar en la empresa. Se analizarán cada uno de los puntos imprescindibles de la empresa y los procesos actuales, así como de las herramientas que se están utilizando actualmente para una mejor selección de la metodología a implementar.
- Familiarización con la nueva herramienta. Al ser una herramienta nueva, habrá que revisar cada una de las funcionalidades que nos proporciona para sacar el máximo rendimiento de ella.
- Selección de la metodología. Junto con los datos extraídos, decidir y argumentar que metodología usar.

Los recursos necesarios para la realización del análisis de este proyecto son:

- **Recursos materiales:** Ordenador.
- **Recursos humanos:** Analista y gestor de proyectos.

### 5.3. IMPLEMENTACIÓN

Una vez seleccionada la metodología con la cual se va a trabajar se va a proceder al diseño de esta y la adaptación de esta a Develoop Software y a las herramientas de trabajo. Este grupo de tareas engloba:

- Definición y detalle de las fases en la gestión de un proyecto. Se van a concretar cada una de las fases de la metodología, así como las tareas que se comprenden en cada una de ellas.
- Adaptación a la nueva herramienta. Una vez definidas las fases y haberse familiarizado con la herramienta, se va a definir el uso que se va a hacer de ella durante la gestión de un proyecto software.

Es el punto clave del proyecto, ya que una vez decidida la metodología, un buen grado de detalle de cada una de las fases que conforman la gestión de un proyecto desembocará en un éxito en el uso de esta.

Los recursos que se usarán en este grupo de tareas serán:

- **Recursos materiales:** Ordenador.
- **Recursos humanos:** Gestor de proyectos.

#### **5.4. FASE FINAL**

Este punto comprende la presentación del TFG para su posterior defensa delante de un tribunal. En esta tarea es muy importante mostrar y explicar los puntos clave del TFG de tal manera que se tenga una visión general pero clara de este trabajo.

Los recursos que se estiman usar en la fase final del proyecto son los siguientes:

- **Recursos materiales:** Ordenador.
- **Recursos humanos:** Gestor de proyectos.

## 6. PLANIFICACIÓN

Una planificación temporal realista y precisa de cada una de las tareas a realizar en el proyecto ayudará en cada momento del proyecto a tener una visión de los puntos que quedan por realizar y a cuanto estamos de la finalización del trabajo.

### 6.1. ESTIMACIONES

Una vez identificadas las distintas tareas a realizar en este TFG, se puede proceder a la estimación, en horas, de cada una de ellas para posteriormente plasmar en un diagrama de Gantt los *timings* de cada una de las tareas y por lo tanto la planificación temporal del proyecto.

ID	Descripción	Tiempo (Horas)	Dependencia
GESTIÓN DEL PROYECTO		76	
T1	Herramientas TIC	5.75	
T2	Contextualización y alcance	24.5	
T3	Planificación temporal	8.25	T2
T4	Presupuesto y sostenibilidad	9.25	T3
T5	Entrega final	18.25	T4
T6	Reuniones	10	
ANÁLISIS		185	
T7	Estado del arte	90	
T8	Empresa	30	
T9	Familiarización con la herramienta	25	
T10	Selección de la metodología	40	T7,T8,T9
IMPLEMENTACIÓN		220	Análisis
T11	Definición y detalle	140	
T12	Adaptación a la herramienta	80	T11
FASE FINAL		20	Implementación
T13	Defensa final	20	
TOTAL		501	

Tabla 1. Estimación de las tareas. Elaboración propia.

### 6.2. DIAGRAMA DE GANTT

De las estimaciones sacadas en el punto anterior se ha generado el siguiente diagrama de Gantt, en el cual se muestra la dedicación a cada una de las tareas identificadas, así como las dependencias entre ellas. En el caso de las reuniones estas se pueden realizar de forma paralela a las otras tareas.

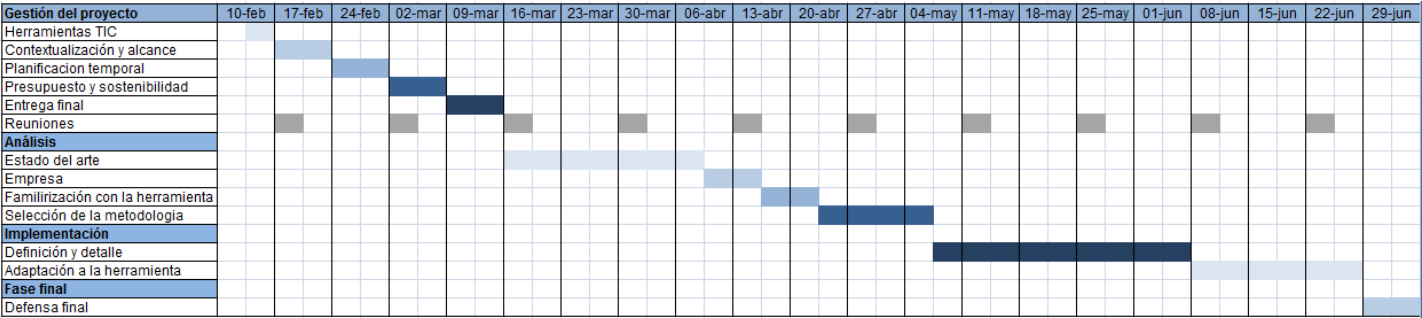


Tabla 2. Diagrama de Gantt del proyecto. Elaboración propia.

## **7. GESTIÓN DE LOS RIESGOS**

En este punto conviene recordar los principales riesgos identificados en el punto “Riesgos y Obstáculos” los cuales fueron:

- El tiempo.
- Mala planificación del proyecto.
- Inexperiencia en el uso de la nueva herramienta.

Para cada uno de los riesgos anteriores, se han elaborado planes alternativos para poder mitigarlos.

### **7.1. PLANES ALTERNATIVOS**

#### **7.1.1. EL TIEMPO**

Como bien comentamos anteriormente, el tiempo es un problema en cualquier proyecto ya que hay una fecha en la que se tiene que entregar, que es inamovible.

Con tal que el tiempo no suponga un problema, se realizarán reuniones con el director del TFG cada dos semanas para realizar un seguimiento del avance del trabajo y poder identificar posibles retrasos.

#### **7.1.2. MALA PLANIFICACIÓN**

La mala planificación de un proyecto, va muy ligado con el problema del tiempo, ya que nos podemos encontrar con tareas que preveíamos que serían mucho más fáciles, nos involucrarían menos tiempo, al hora de realizarlas han resultado más complejas de lo esperado.

Es por eso que en la planificación temporal, se han estimado las tareas a la alza, para cubrir posibles complicaciones en la realización de estas.



### **7.1.2. INEXPERIENCIA EN EL USO DE LA NUEVA HERRAMIENTA**

La familiarización con la herramienta se ha previsto realizar de forma individual y con ayuda de video explicativos que Odoo provee a sus *partners*.

Si se llegara a detectar que la familiarización de esta herramienta toma más tiempo de lo requerido, se pedirá ayuda a compañeros que ya la han utilizados para agilizar este proceso.

## 8. PRESUPUESTO

### 8.1. IDENTIFICACIÓN DE COSTES

Para la realización de un buen presupuesto es necesario identificar cada uno de los recursos que se van a destinar a este trabajo, ya sean recursos humanos, es decir, personas que van a contribuir a la realización de este, como recursos hardware y software e incluso aquellos costes indirectos como pueden ser la luz, el agua, internet, entre otros que especificaremos a continuación.

#### 8.1.1 COSTES HUMANOS

En este trabajo van a contribuir dos personas una va a ser el director de este TFG, el cual me ayudará en la toma de decisiones a la hora de escoger la metodología y en enfoque de algunos de los puntos, así como de la gestión del trabajo (Todas estas horas recogidas en la tarea de “Reuniones”), y un servidor que se encargara del análisis y también de la propia gestión, así como de la implantación de la metodología.

Para definir con un buen grado de detalle las horas dedicadas en el trabajo y el rol que se va a desempeñar, recogeremos las tareas identificadas en puntos anteriores.

ID	Descripción	Horas	Roles	Coste(€)
GESTIÓN DEL PROYECTO				927
T1	Herramientas TIC	5.75	Project Manager	54.625
T2	Contextualización y alcance	24.5	Project Manager	232.75
T3	Planificación temporal	8.25	Project Manager	78.375
T4	Presupuesto y sostenibilidad	9.25	Project Manager	87.875
T5	Entrega final	18.25	Project Manager	173.375
T6	Reuniones	10	2x Project Manager	300
ANÁLISIS				1757.5
T7	Estado del arte	90	Analista	855
T8	Empresa	30	Analista	285
T9	Familiarización con la herramienta	25	Analista	237.5
T10	Selección de la metodología	40	Project Manager	380

IMPLEMENTACIÓN				2090
T11	Definición y detalle	140	Project Manager	1330
T12	Adaptación a la herramienta	80	Project Manager	760
FASE FINAL				190
T13	Defensa final	20	Project Manager	190
TOTAL				4964.5

**Tabla 3. Costes humanos. Elaboración propia.**

El salario bruto asignado a mi persona son de 9.5€/h es el precio acordado para el convenio actual entre la empresa y la universidad para estas prácticas curriculares. En cambio al director le asignaremos el salario de un IT Project Manager Sénior que según *Glassdoor*, se sitúa en unos 20.5€/h brutos.

A este coste final hay que incluirle el coste de la seguridad social por lo que:

$$\text{PRECIO TOTAL} = 4964.5 * 1.3 = 6453.85$$

### 8.1.2. COSTES HARDWARE

Como costes de hardware en este trabajo únicamente tenemos aquello con lo que vamos a redactar el TFG. Y todos ellos se van a usar en la totalidad de las horas del proyecto:

Hardware	Precio	Vida útil	€/mes	Tiempo	Coste (€)
HP Probook 450 G4	760	5 años	12.66	5 meses	63.33
Sync Master 720N	35	5 años	0.583	5 meses	3
Logitech B10	5	5 años	0.083	5 meses	0.41
Logitech K12	16	5 años	0.267	5 meses	1.33
TOTAL	-	-	-	-	68.07

**Tabla 4. Costes de hardware. Elaboración propia.**

### 8.1.3. COSTES SOFTWARE

Además del uso de hardware en este trabajo también se van a utilizar una serie de programas los cuales debemos considerar para el cálculo de los costes:

Software	€/mes	Tiempo	Coste (€)
Office 365 Empresa	8.80	5 meses	44
Odoo - Módulo proyectos	12	5 meses	60
Trello	0	5 meses	0
MandaHoras	0	5 meses	0
TOTAL	-	-	104

Tabla 5. Costes de software. Elaboración propia.

### 8.1.4. COSTES GENÉRICOS

Existen otro tipo de costes, que de manera indirecta intervienen en este proyecto, por lo que también los vamos a tener en cuenta.

Producto	€/mes	Tiempo	Coste (€)
Electricidad	500	5 meses	2500
Agua	370	5 meses	1875
Oficina	1400	5 meses	7000
Internet	180	5 meses	900
Transporte	20	5 meses	100
TOTAL	-	-	12375

Tabla 6. Costes genéricos. Elaboración propia.

## 8.2. CONTINGENCIAS

En todo proyecto existe un margen de seguridad en los costes cuantificados para posibles imprevistos durante el proyecto. Es por ese que en este caso en concreto fijaremos un 5% de margen a cada una de las partidas detalladas anteriormente:

Partida	Coste (€)	Contingencia	Coste final (€)
Coste humano	6453.85	5%	6776.54
Coste hardware	68.07	5%	71.47
Coste software	104	5%	109.2
Costes genéricos	12375	5%	12993.75
TOTAL	-	-	19950.96

**Tabla 7. Costes finales aplicando contingencias. Elaboración propia.**

### 8.3. CONTROL DE GESTIÓN

En apartados anteriores se han comentado los riesgos que existen en el trabajo y la manera en que se iban a gestionar para que esto no sucediera.

En este caso los únicos imprevistos que pueden aumentar un coste en el presupuesto real final son aquellos que provengan del tiempo. En este caso en concreto, se han aplicado márgenes tanto al coste final del presupuesto, como al tiempo dedicado a cada una de las tareas, hecho que nos ayudará a mantener un presupuesto controlado.

A parte de los márgenes para tener un control real de los costes de este TFG, cada vez que se acabe un grupo de tareas se realizará un control exhaustivo de las horas dedicadas a cada una de las tareas y se compararan con las horas estimadas y junto con estos datos se calcularán las desviaciones, siguiendo las siguientes formulas:

**Desviación en precio:**  $(\text{Coste estimado} - \text{Coste real}) / \text{Horas reales}$

**Desviación en consumo:**  $(\text{Horas estimadas} - \text{Horas reales}) * \text{Coste estimado}$

**Desviación total:**  $\text{Desviación en precio} + \text{Desviación en consumo}$

Como hemos comentado anteriormente las únicas desviaciones que calculamos son aquellas referentes al tiempo que repercuten a los costes humanos, para los costes genéricos como de software o hardware no hay posibles desviaciones considerables por lo que no las tendremos en cuenta.

## **9. INFORME DE SOSTENIBILIDAD**

Tras realizar la autoevaluación sobre la sostenibilidad de los proyectos, procederemos a analizar las conclusiones sacadas.

La autoevaluación me hace ver que aunque conozca bien los efectos medioambientales y todo lo referente a la sostenibilidad en la realización de un proyecto software, no los aplico a ninguno de mis proyectos, ni incluso los tengo en cuenta a la hora de la realización de este por lo que no se incluyen indicadores. Creo que esto debería cambiar, ya que tras darme cuenta del impacto del software en la sostenibilidad de la sociedad actual debería tenerse en cuenta en todos y cada uno de los proyectos para que realmente todos pongamos en valor y conozcamos el grado de impacto de cada proyecto y entre todos cambiemos esta dinámica de no tener en cuenta los aspectos sociales, ambientales...

En referencia a este proyecto, personalmente, creo que este puede tener un buen impacto en cada una de las dimensiones que detallaremos posteriormente no únicamente de cara a la empresa sino también de cara al cliente.

### **9.1. DIMENSIÓN ECONÓMICA**

Teniendo en cuenta el coste total de este proyecto, y valorando el impacto que este puede llegar a tener en la empresa, podemos afirmar que no es un coste desmesurado ya que la implantación de una metodología puede ayudar a optimizar tiempo y aumentar el rendimiento de cada uno de los implicados de un proyecto y eso desemboca en un ahorro de dinero.

Al utilizar una solución existente, se valorará el impacto de cada uno de las metodologías propuestas y como estas ayudan, actualmente, a las empresas que las están utilizando.

## **9.2. DIMENSIÓN AMBIENTAL**

La dimensión ambiental de este proyecto no se ha estimado dado que no tiene un gran impacto ambiental únicamente es una mejora a nivel procedimental en la gestión de un proyecto por lo que no se puede ni minimizar el impacto ambiental y mucho menos mejorar soluciones ya existentes, ya que lo que se va a realizar va a ser una adaptación de una solución.

Si tuviéramos que resaltar algún impacto ambiental sería el consumo eléctrico y de internet, que no podemos reducir de ninguna manera

## **9.3. DIMENSIÓN SOCIAL**

Personalmente, la realización de este proyecto me permitirá aumentar mi conocimiento en el área de gestión de proyectos un únicamente conociendo las metodologías actuales, sino también pudiendo analizar los puntos fuertes y puntos débiles de cada una de ellas, y saber cuándo es posible utilizar una o cuando utilizar otra.

Como bien hemos dicho anteriormente, nosotros no crearemos una nueva metodología, lo que haremos va a ser adaptar una de las ya existentes, por lo que a nivel social no mejora respecto a las existentes ya que partimos desde una solución ya implementada. Lo que si va a aportar a los integrantes de la empresa una mayor decisión en las tareas a realizar y ayudará a aumentar el rendimiento del equipo de trabajo.

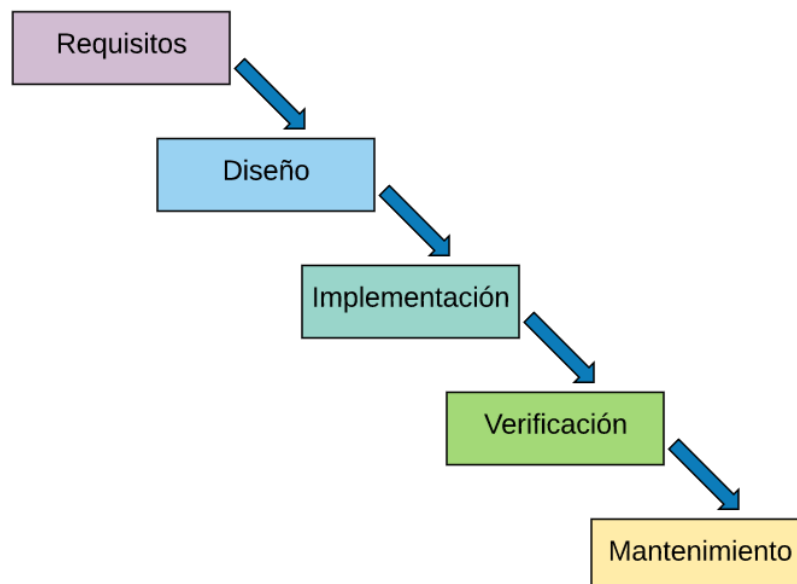
## 10. ESTADO DEL ARTE

Una vez finalizada la primera fase, más específica de la gestión de proyectos, procederemos a investigar algunas de las metodologías que existen hoy en día que creemos que pueden encajar mejor en la empresa. Analizaremos su funcionamiento así como los puntos fuertes y débiles de cada una de ellas. De esta manera la en la futura decisión final se podrán tener en cuenta todos y cada uno de los puntos de cada metodología.

Primero de todo empezaremos con la explicación de los dos grandes grupos de metodologías que existen hoy en día: las metodologías tradicionales o en cascada y las metodologías ágiles.

### 10.1. METODOLOGIA TRADICIONAL O EN CASCADA

La metodología tradicional o en cascada es un modelo lineal que emplea un diseño secuencial en el desarrollo de las distintas etapas en el que este se compone. Estas etapas se ejecutan una tras otras, es decir siguiendo el flujo de arriba abajo, como una cascada.



**Ilustración 1. Etapas en metodología en cascada.**

**Recuperado de:** <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>



Como se puede apreciar en la figura anterior, el modelo tradicional se divide en 5 fases. No se puede empezar una etapa hasta que no se haya terminado la anterior.

- **Fase de requisitos.** Se describe la estructura interna del software y las relaciones entre cada una de las entidades, así como la definición de los problemas complejos y las tareas secundarias que componen todos y cada uno de estos.

- **Fase de diseño.** Es la fase en la que se diseña la arquitectura del software haciendo especial hincapié en los componentes y entornos de trabajo.

- **Fase de implementación.** Se incluye el desarrollo/programación del software haciendo uso de lo acordado en la fase de diseño.

- **Fase de verificación.** Fase en la que se comprueban que todos los componentes del software funcionen correctamente y cumplan con los requisitos y funcionalidades captadas en la primera etapa.

- **Fase de mantenimiento.** Una vez realizado la implementación y todas las verificaciones se instala la aplicación en el entorno de producción y se comprueba que funcione correctamente. Se debe asegurar un correcto funcionamiento a lo largo del tiempo.

Existen ampliaciones de este modelo en cascada en el que se añaden algunas funciones iterativas, como por ejemplo la comprobación de los resultados de la fase anterior.

[18] [19] [20]

## 10.2. METODOLOGIA ÁGIL

Como alternativa a los métodos tradicionales surgieron los métodos ágiles, caracterizados principalmente por la rapidez y flexibilidad. Estas dividen el proyecto en pequeñas tareas que deben acabarse y entregarse en iteraciones de tiempo generalmente cortas.

Estas metodologías nacen con el fin de satisfacer la demanda del cliente, que normalmente es cambiante durante el desarrollo del proyecto y que deben adaptarse a estas sin tener que empezar desde cero el proyecto. Por lo que el cliente está bastante implicado en el proyecto, es por eso que al fin de cada iteración se realizan entregas parciales al cliente para su verificación.

Una de las principales características de las metodologías ágiles es la constante interacción con el cliente, que sirven para eliminar o detectar cualquier funcionalidad no requerida o un enfoque erróneo.

[21] [22]

### 10.3. SCRUM

La metodología Scrum es una de las posibles implementaciones de las metodologías ágiles. Uno de los principios claves del éxito de Scrum, se basa en la premisa que, durante el desarrollo de producto los clientes podrán cambiar sus opiniones sobre lo que quieren y lo que necesitan. Los imprevistos no son una gran dificultad para la implementación ya que se acepta que el problema no está completamente definido y se centra en la respuesta rápida y responder a las necesidades de última hora. [10]

En Scrum existen dos artefactos bien definidos para que todo el equipo tenga una misma visión de lo que hay en el proyecto. Estos dos artefactos son los siguientes:

- Product Backlog. Es el listado de todas las tareas en las que se ha dividido todo el proyecto. Cualquier cosa que se deba hacer en la realización del proyecto debe estar definida en el *Product Backlog* y estimada correctamente. Cada tarea debe tener definida una prioridad.

- Sprint Backlog. Es el conjunto de tareas del *Product Backlog* que se escogen para desarrollar en cada sprint. Deben quedar bien definidos y ser conocidos por todo el equipo para focalizarse en la realización de estas.

En el proceso de la gestión de un proyecto en Scrum se divide en *Sprints* que cada uno de ellos está compuesto por los siguientes pasos:

- *Sprint planning*. Es la primera reunión que se realiza en el *Sprint*, en la que se eligen las tareas del *Product Backlog* que se van a realizar en la iteración (*Sprint Backlog*) así como el objetivo de esta (*Sprint Goal*).

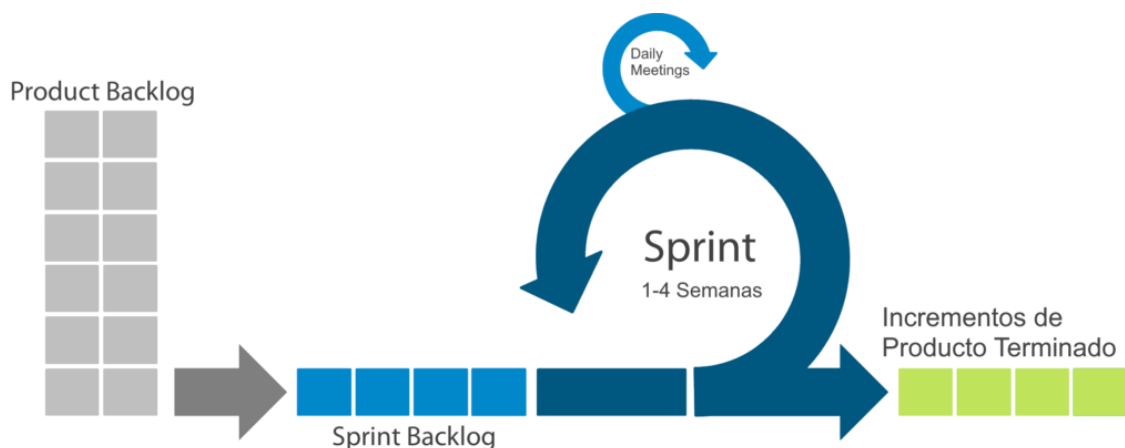
- *Daily meeting*. Reunión diaria dentro del *Sprint* en la que se responden la siguientes 3 preguntas:

- 1) Que tareas completé ayer
- 2) Que tareas voy a realizar hoy
- 3) Impedimentos que no me han permitido avanzar.

Esta reunión permite poder adaptarse si se ha realizado algún cambio de las tareas dentro de la iteración.

- *Sprint review*. En esta reunión se presenta al cliente lo realizado en el *Sprint* y en la que se muestra su funcionamiento. El cliente valida lo realizado a parte de aportar el *feedback* sobre posibles nuevas tareas a realizar dentro del proyecto.

- *Sprint retrospective*. Es el último paso de la iteración en la que se hace una auto-evaluación de cómo ha ido el *Sprint*, que se debería mejorar y que cosas se deberían empezar a realizar o dejar de hacer en los siguientes *Sprints*.



**Ilustración 2. Flujo en Scrum.**

Recuperado de: <https://richardgracia.com/scrum-para-startups/>

En todo equipo Scrum debe haber una persona que desempeñe el rol de **Producto Owner**, encargado de darle un valor añadido al software ya que debe conocer el negocio y va a ser él mismo el encargado de tomar las decisiones sobre la dirección que va a tomar el desarrollo [7], y el de **Scrum Master** responsable de que se sigan las prácticas y valores descritos en el modelo Scrum.[8]

Las principales ventajas del uso de la metodología Scrum son las siguientes:

- Mayor eficiencia, incluso desde el punto de vista económico.
- Mayor visión de todos los miembros del equipo de desarrollo.
- Promueve una participación activa por parte de todo el equipo, permite obtener un producto más aproximado a las necesidades reales.

Por el contrario, también encontramos una serie de desventajas al usar esta metodología en los proyectos:

- Una mala definición de una tarea deriva en inexactitud en la estimación de los costes y el tiempo del proyecto.
- Sin el compromiso del equipo, es difícil que el proyecto finalice de manera exitosa.

[9]

#### 10.4. PRINCE2

PRINCE2 (*Project In Controlled Enviroment*) es un método para la dirección de proyectos en la que se manejan una carga importante de variabilidad y de incertidumbre en entornos controlados. Es una metodología estructurada y flexible que permite manejar la administración y la gestión de los proyectos de cualquier tamaño.

Esta metodología se apoya en 7 principios que no únicamente enrique al proyecto que se desarrolla sino a todos los implicados en la implementación de este:

- **Justificación comercial continúa.** Hay que asegurar que hay un motivo que justifique el inicio del proyecto y que este se siga respetando a lo largo del proyecto (Business Case).

- **Aprender de la experiencia.** Se debe tener en cuenta todo lo aprendido de experiencias anteriores, las que se obtienen durante el proceso y con el fin de este.

- **Roles y responsabilidades definidos.** Un proyecto debe tener definidos y acordados los roles y responsabilidades.

- **Gestión por fases.** El proyecto con tal de manejarse de una manera más fácil se divide en fases. En PRINCE2 se planifica, se supervisa y se controla fase por fase. Cada fase se separa de la siguiente por “puntos de control”, en el cual se evalúa el desempeño de la fase, se decide si se continúa con la siguiente fase y se realiza el plan para esta.

- **Gestión por excepción.** En PRINCE2 las cuestiones importantes se conocen como excepciones. Se establecen equipos de referencia en los que se delegan la gestión de las actividades diarias del proyecto. Si surge cualquier problema (Excepción), se podrá consultar a los superiores como actuar.

- **Orientación a productos.** Hay que concentrarse en la definición y entrega de productos. El proyecto no se define como un conjunto de tareas, sino que se entregan productos, que se elaboran tras la ejecución de las tareas.

- **Adaptación.** La metodología a usar y los controles que se aplican sobre el proyecto se basan en su tamaño, complejidad y nivel de riesgo.

Cada uno de los procesos que aparecen en PRINCE2, contribuyen al objetivo principal, que no es otro que el éxito del proyecto. En PRINCE2 hay 8 procesos, aunque a veces se consideran 7:

**1. Empezar el proyecto.** En este proceso se determina si el proyecto es viable, por lo que se decide si se debe continuar con la siguiente fase. Por lo que una vez finalizado este proyecto se debe tener claro e implementado: el Business Case, el enfoque del proyecto, el equipo de personas que realizarán

el trabajo, el expediente del proyecto y el plan de fase a realizar en la siguiente fase.

**2. Dirigir el proyecto.** Se dirigen las responsabilidades en la supervisión del proyecto. Se proporcionan los protocolos de aprobación al final de cada etapa y en el cierre del proyecto. Es el marco en el que se reciben los requisitos y funcionalidades del proyecto.

**3. Iniciar el proyecto.** En él se plantean como se puede gestionar el proyecto y se plasma en el PID (*Project Initiation Document*) con el objetivo de que se establezca unos puntos en los que se acuerden los elementos críticos del proyecto, así como para la primera etapa de desarrollo del proyecto.

**4. Planificación.** En los planes se identifican los entregables de los productos, las actividades y los recursos asignados para crearlos todo ello siendo consistente con lo descrito en el PID. (Este es el proceso que no se considera en algunos casos)

**5. Control de etapa.** En este proceso se realiza una guía para la gestión diaria del proyecto. Para cada etapa del proyecto se realiza este control.

**6. Gestión de entrega del producto.** Es el mecanismo que sirve para que el equipo de desarrollo acuerde los trabajos a realizar, y esto es repetido por cada bloque de trabajo.

**7. Gestión de los límites de etapa.** Se garantiza que el trabajo que se ha finalizado se ha realizado acorde a los requisitos establecidos.

**8. Cerrar el proyecto.** El cierre del proyecto puede deberse a una finalización exitosa del trabajo como de una finalización prematura. En el caso que haya sido una finalización satisfactoria hay que garantizar que este satisface las necesidades del cliente.

## The PRINCE2 Process Model Diagram

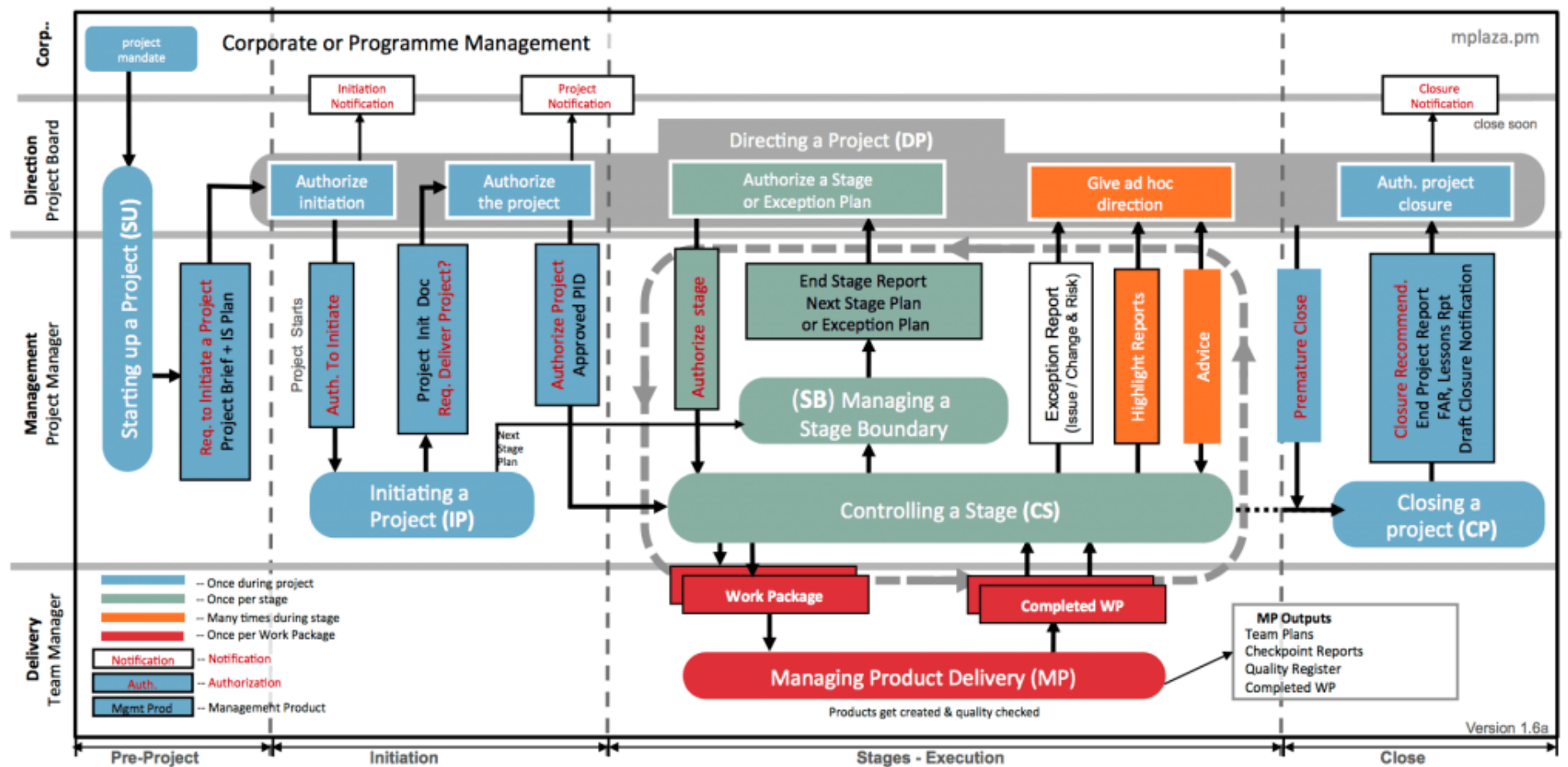


Ilustración 3. Proceso en PRINCE2.

Recuperado de: <https://adictec.com/que-es-prince2/>

En última instancia, hay distintas aéreas de gestión que deben ser gestionadas a lo largo de todo el proyecto. Estas aéreas de gestión, también llamadas temáticas, son actividades que tienen lugar al inicio del proyecto y a lo largo de este. Proporcionan una orientación de la manera en la que se deben realizar las cosas durante el proyecto.

Las temáticas deben adaptarse al proyecto sobre el que se está trabajando y dar énfasis a aquellas que sean más relevantes según el tipo de proyecto. Las temáticas son:

- **Business Case.** Hay que establecer un motivo para juzgar si el proyecto es deseable, viable y alcanzable.
- **Organización.** Describe las funciones y responsabilidades del equipo de gestión del proyecto.
- **Calidad.** Implementar un sistema para verificar que los productos son aptos para uso, es decir que alcanzaran las expectativas y que podrán ser utilizados como se pretendía.
- **Planes.** Se concretan los pasos necesarios para entregar los distintos productos y aquellas técnicas que se deben aplicar, así como la comunicación a lo largo de todo el desarrollo.
- **Controles.** Se encargan de realizar una serie de verificaciones que pretendan hacer notorio si se ajusta a todo lo planificado.
- **Riesgos.** Se pretende llevar un control de los planes alternativos a aquellos obstáculos que van apareciendo a lo largo del proyecto.
- **Cambio.** En esta temática se calcula el impacto de los cambios potenciales: su importancia, costes adicionales, impacto en el negocio.
- **Progreso.** Explica el proceso de la toma de decisiones para la aprobación de los planes y la supervisión del proyecto. Determina si el proyecto debe continuar y cómo. [5] [11]

Las ventajas del uso de PRINCE2 para la gestión de los proyectos son:



- Todos los participantes del proyecto tienen claro su rol y lo que se espera de ellos, ya que los roles y las responsabilidades están claramente definidos.
- La estructura de PRINCE2 es suficientemente sencilla que permite una adaptación a las particularidades de cada proyecto.
- Control completo de recursos y riesgos así como de la organización de comienzos, centros y finales. [12]

No todo son puntos positivos, también existen algunas desventajas a tener en cuenta:

- Todos los puntos requeridos tienen que estar bien definidos para que la metodología proporcione todo lo que se espera de ella. Todo debe estar controlado.
- Existe un número muy reducido de herramientas y técnicas para el control de la metodología. [13]

## 10.5. PMBOK

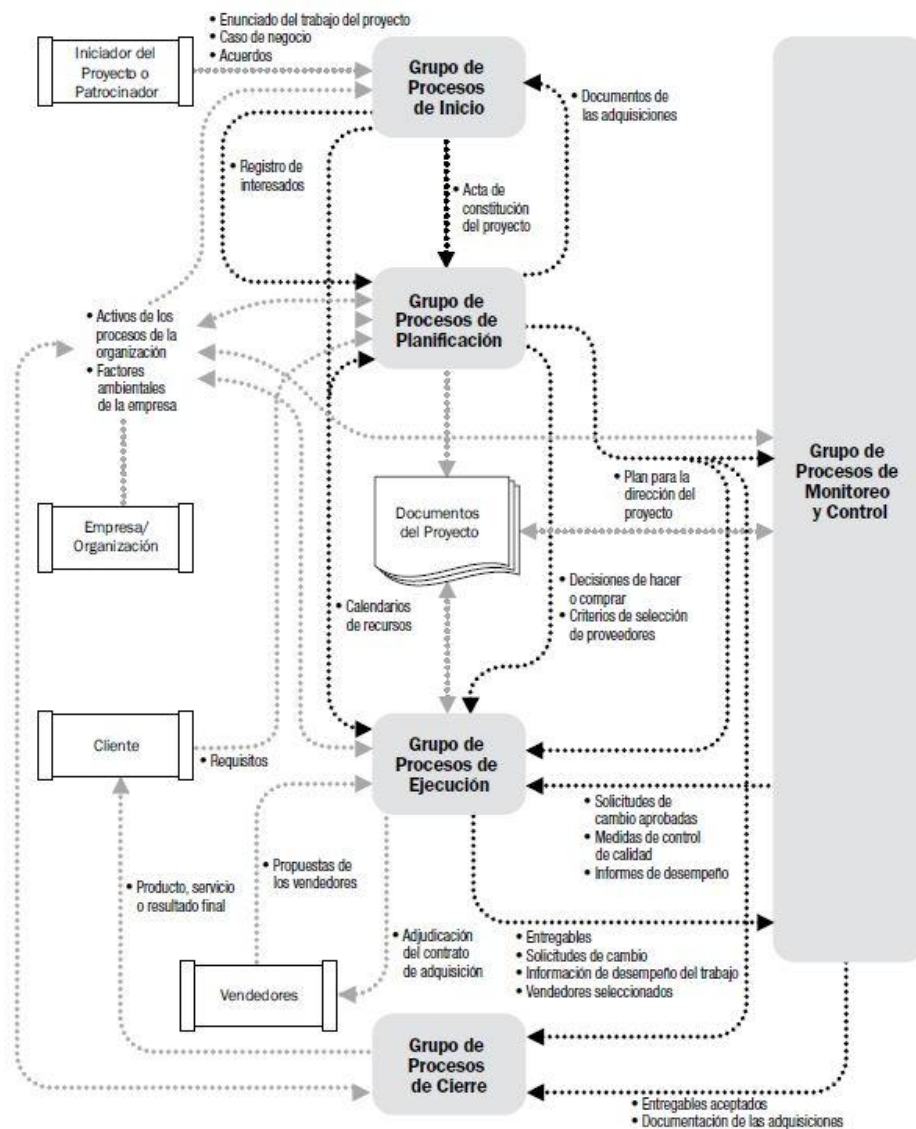
La guía PMBOK es un instrumento desarrollado por el *Project Management Institute* (PMI) en la cual se concretan una serie de buenas prácticas relacionadas con la dirección de los proyectos con la implementación de técnicas y herramientas que permiten la identificación de 47 procesos que se dividen en 5 grupos y 10 áreas de conocimiento.

A continuación se explican los 5 grupos de procesos:

- 1. Inicio.** En el cual se engloban 2 procesos con el objetivo de definir un nuevo proyecto o una fase de un proyecto ya existente.
- 2. Planificación.** Se Incluyen 24 procesos los cuales ayudan a concretar, establecer objetivos y definir el alcance del proyecto y de qué manera se llegara a lograr la consecución de estos.
- 3. Ejecución.** Definido por 8 procesos destinados a completar todo aquello que se ha definido en la planificación.

**4. Monitorización y control.** En este macroproceso se incluyen 8 procesos, relacionados con el seguimiento de la correcta realización del desarrollo del proyecto, así como de su revisión y control.

**5. Cierre.** Formado por dos procesos que se encargan de cerrar el proyecto y finalizar las tareas de todos los procesos.



**Ilustración 4. Procesos PMBOK. Recuperado de:**  
<http://www.ingenieriasystems.com/2016/06/grupos-de-procesos-de-la-direccion-de-proyectos-de-inicio-PMBOK.html>

En cada uno de los procesos comentados intervienen 10 áreas de conocimiento que se enuncian y se describen en la guía PMBOK de la siguiente manera:

**1. Integración.** Relacionada con la dirección de proyectos, en la que se establecen los criterios para una correcta dirección de los distintos procesos y actividades: gestión, administración y coordinación.

**2. Alcance.** Determine el alcance del proyecto, definiendo todas las actividades y procesos implicados en él.

**3. Plazos.** Gestión del tiempo de los procesos definidos en el alcance del proyecto, y gestión de ellos para su control.

**4. Costes.** Procesos necesarios para planificar, estimar y presupuestar los costes para que el proyecto se pueda finalizar con lo estimado a nivel económico.

**5. Calidad.** Determinar las políticas de calidad para la evaluación de los resultados conseguidos.

**6. Recursos Humanos.** Organizar y gestionar el equipo de proyecto definiendo responsabilidades.

**7. Comunicación.** Responsable de la gestión y administración de las informaciones y las vías de comunicación entre las distintas áreas del proyecto así como la elaboración de información del proyecto orientada al exterior.

**8. Riesgos.** Detección y gestión/control de los riesgos implicados en cada uno de los procesos y fases, así como de las medidas a adoptar si estos aparecen.

**9. Adquisiciones.** Gestión de procesos de compra de todo aquello necesario para los distintos equipos implicados en el desarrollo del proyecto: estructuras, herramientas, servicios.

**10. Stakeholders/Implicados.** Identificación y gestión de todo aquel implicado en el proyecto, así como el impacto en cada uno de ellos.

[5] [14]

Habiendo definido PMBOK nos encontramos con puntos positivos a resaltar en la utilización de esta metodología en la gestión de un proyecto:

- Hay un apartado dedicado para la detección y control de los riesgos del proyecto, y planes alternativos si estos aparecen.
- Idóneo para proyectos con requisitos bien definidos.
- Alto grado de detalle del alcance, costes y *deadlines* de cara al cliente.

Por lo contrario, existen algunos puntos negativos en contra que hay que tener en cuenta:

- No se adapta a aquellos proyectos con cambios constantes a nivel funcional ya que pueden desembocar en desajustes considerables en la planificación y en los costes.
- Cualquier proceso no detallado correctamente, como puede ser aquellos relacionados con la dirección puede crear incertidumbre.

## **10.6. AGILE UNIFIED PROCES (AUP)**

Es una versión simplificada de Rational Unified Process (RUP). Describe una manera simple de entender el desarrollo de software usando técnicas ágiles y conceptos que provienen de RUP. En estas técnicas ágiles se incluyen el TDD (Test DrivenDevelopment), Agile Model Driven Development (AMDD), gestión de cambios ágiles y refactorización de la base de datos para mejorar la productividad.

En la siguiente figura se muestra de manera grafica el ciclo del desarrollo usando AUP. Como podemos ver se compone de 4 fases y 7 disciplinas, las cuales tendrán una mayor presencia dependiendo en la fase en la que nos encontremos.

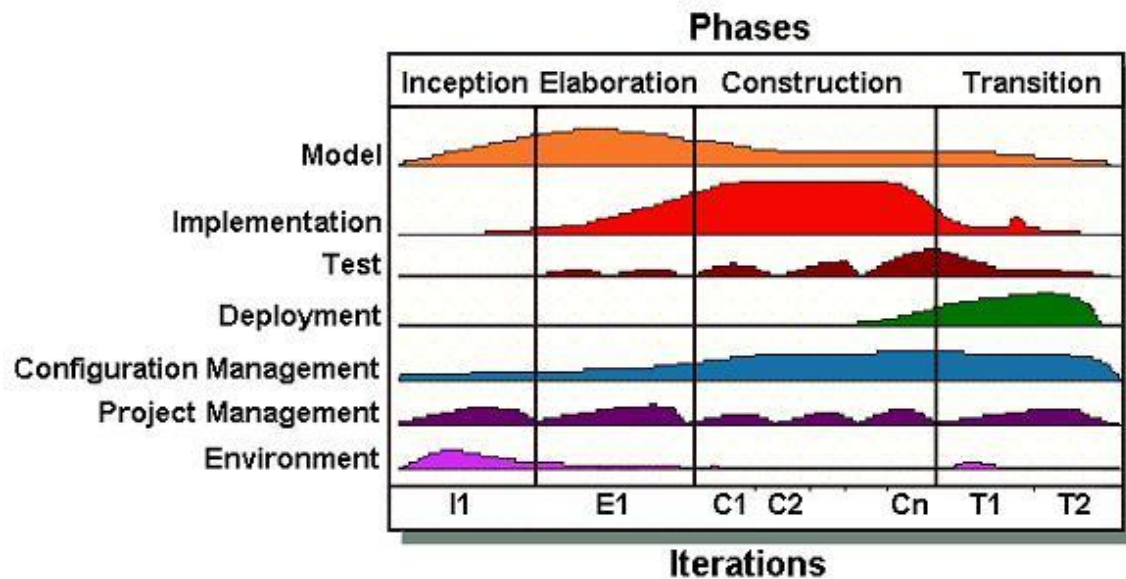


Ilustración 5. Fases en AUP.

Recuperado de: [https://www.ecured.cu/Agile\\_Unified\\_Process](https://www.ecured.cu/Agile_Unified_Process)

Primeramente, empezaremos definiendo las fases de AUP:

- **Inicio.** Se identifica el alcance del proyecto, la arquitectura del sistema, obtención de fondos y la aceptación de los implicados.
- **Elaboración.** El objetivo principal es probar la arquitectura del sistema.
- **Construcción.** Se desarrolla el software de forma incremental cumpliendo con las prioridades y requisitos de los *stakeholders*.
- **Transición.** Validación y puesta en marcha del sistema en el entorno de producción.

A continuación definiremos las disciplinas que se encuentran en las fases descritas anteriormente:

- **Modelado.** El objetivo principal es la comprensión del modelo de negocio de la organización, el problema que pretende resolver el *software* y la solución más factible.
- **Implementación.** Se transforma la solución detectada en la fase de modelado en un código ejecutable y se realizan las pruebas básicas de su correcto funcionamiento.

- **Test.** Asegurar la calidad del software, así como detección de errores y validación del sistema y los requisitos. Como se puede ver en el gráfico mostrado con anterioridad, se realiza en paralelo a la implementación del código.
- **Despliegue.** Se entrega el proyecto y se despliega en el entorno de producción para que este esté operativo para los usuarios.
- **Gestión de la configuración.** Gestión del acceso de los entregables del software. En la que se incluyen el control de versiones y la gestión de los cambios.
- **Gestión del proyecto.** Dirección de las tareas que se desarrollan en el proyecto. Gestión de los riesgos y coordinación del equipo de trabajo de tal manera que se cumpla con el tiempo y el presupuesto estimado.
- **Entorno.** Aseguración que los procesos óptimos, las guías, los estándares y las herramientas estén disponible para el equipo. Haciendo referencia al gráfico se puede ver que esta disciplina toma mayor importancia al inicio para asegurarse que se puede proceder con las siguientes fases.

[15] [16]

Teniendo nociones básicas de cómo funciona AUP, se detectan las siguientes ventajas al uso de esta metodología:

- Todo el mundo sabe lo que tiene que hacer.
- Simplicidad en la documentación. Información relevante documentada en las páginas necesarias y no más de la cuenta.
- Se realiza la comprobación del código (tests), a medida que se realiza la implementación del código por lo que cualquier bug puede ser detectado y resuelto sin afectar demasiado a lo ya realizado.

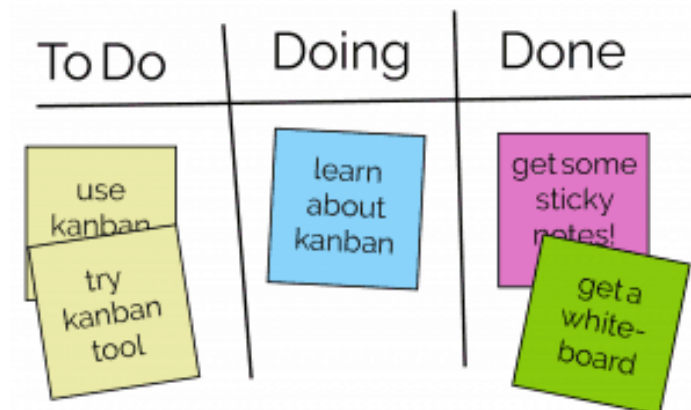
Por otro lado tenemos algunos puntos negativos a tener en cuenta, como pueden ser:

- Al ser un proceso simplificado del RUP no se disponen de mucho detalles en el proceso por lo que puede ser un hándicap.
- Como cualquier metodología ágil, no es optimo para iteraciones muy largas.

## 10.7. KANBAN

El principal objetivo de Kanban, que pertenece al grupo de las metodologías ágiles, es obtener de una manera visual el estado de un proyecto así como las prioridades, los tiempos, los estados de las tareas y la carga de trabajo de los integrantes del equipo. Para esto el proyecto se divide en distintas partes, que normalmente se escriben en una tarjeta y se ponen en un tablero. La tarjeta debe estar compuesta por distinta información como puede ser la descripción o la estimación temporal.

El tablero se divide en distintas columnas, una para cada estado posible de una tarea. Por lo que las fases de los ciclos o flujo de trabajo se deciden según el caso, por lo que es una de las primeras cosas que debemos definir en el proyecto.



**Ilustración 6. Vista Kanban.**

Recuperado de: <http://www.datascience-pm.com/kanban/>

Los principios en los que se diferencia Kanban de otras metodologías son:

- **Calidad garantizada.** No hay margen de error en la tarea por lo que debe salir correctamente de primeras, ya que se basa en que cuesta más después arreglar lo hecho que no hacerlo bien de un primer momento.

- **Reducción del desperdicio.** Se realiza únicamente lo justo y necesario, pero siempre priorizando el buen hacer. Supone una reducción de aquello secundario o no tan prioritario y por lo tanto una reducción del tiempo.

- **Flexibilidad.** Como en toda metodología ágil, se aceptan los cambios por lo que en se decide al principio de cada ciclo que tareas se realizarán a continuación y cuáles son las más prioritarias.

- **Stop starting, start finishing.** Se prioriza acabar el trabajo en curso ya empezado antes que empezar nuevas tareas. Por lo que en cada iteración se definen el máximo número de tareas que se pueden tener en cada fase.

En Kanban no se describen roles, por lo que no hay roles definidos, de esta manera se evita la resistencia de cualquier miembro a realizar un nuevo trabajo. Cabe resaltar que no se tienen fijadas reuniones periódicas, sino que se resalta el flujo en el que se dividen las necesidades del proyecto.

Cabe resaltar los beneficios de gestionar un proyecto utilizando la metodología Kanban:

- Altamente visual. Como hemos comentado de una pasada podemos tener una idea bastante detallada del estado del proyecto y de las tareas.

- Flexible. No únicamente al aceptar nuevas tareas sino al intercambios de tareas entre equipos que hemos visto, al no tener los roles definidos.

- Acabar las tareas antes de empezar nuevas. Antes que ir abriendo tareas e irlas empezando, Kanban prioriza primero acabar las tareas empezadas antes de empezar nuevas, de esta manera se coordinan de una mejor manera las tareas.

Como en cualquier cosa en este mundo existen debilidades las cuales hay que tener en cuenta a la hora de decidir si usar una metodología u otra:



- No existen *deadlines*. Se requiere un alto grado de disciplina del equipo de trabajo para que este no esté desmotivado. Esto es en parte porque se prefiere una tarea bien acabada, sin planificar cuando, que una tarea a corregir en un futuro.
- Dada su naturaleza Kanban es ideal para los sistemas de producción repetitiva. No es una técnica específica del desarrollo software.

[17]

## 11. EMPRESA

Con el fin de escoger la metodología más adecuada para que podamos empezar a usar, debemos tener en cuenta la manera de trabajar actual, el tipo de cliente y el equipo Develoop.

### 11.1. EQUIPO DEVELOOP

En un proyecto existen dos áreas bien distinguidas: desarrollo del proyecto y gestión del proyecto. Es por eso que Develoop está formado por 7 gestores de proyectos y 35 programadores, que abarcan cada una de las tecnologías que ofrece la empresa a sus clientes.

De los 7 gestores de proyectos existen algunos de ellos que desempeñan más de un rol como pueden ser:

- Comercial. Encargado de primer contacto con el cliente y realizar las primeras gestiones del lead.
- Director. Es el encargado de coordinar con cada uno de los gestores las tareas a realizar en cada uno de los proyectos. Junto con el director técnico se encargan de la gestión de los recursos en cada uno de los proyectos presentes.

En el equipo de programación, como bien hemos comentado, existen profesionales con experiencia en cada una de las tecnologías que se ofrecen, y uno de ellos con el rol de director técnico que junto con el director se encarga de la gestión de los equipos.

Como normal general, un equipo de proyecto se compone por:

- Dos *project managers*. Uno que es quien llevará la voz cantante en el proyecto y el otro, como lo llamamos nosotros, va a desempeñar el rol de *backup* con el fin de dar soporte en el área de gestión del proyecto y que debe estar al día en todo lo relacionado con el proyecto.
- Responsable técnico del proyecto. Se encarga de la supervisión técnica del proyecto y si, fuera necesario, del desarrollo de este.

- Dos desarrolladores o programadores. Se encarga principalmente de la implementación íntegra del proyecto.

El número de integrantes de los equipos puede variar según la dimensión del proyecto.

## **11.2. HERRAMIENTAS PARA LA GESTIÓN**

Para una correcta gestión de un proyecto, es necesario tener el soporte de herramientas que faciliten esta gestión. En Develoop actualmente se usan 3: MandaHoras, Excel y Trello.

### **11.2.1. MANDAHORAS**

Este es un programa interno desarrollado por y para Develoop con el fin de contabilizar las horas dedicadas a cada proyecto y poder tener una noción de si se está cumpliendo con lo establecido, además en cada conjunto de horas contabilizadas en un paquete se le añade una descripción de en que se ha estado trabajando.

Además, las horas que contabilizan los desarrolladores, son enviadas a los responsables para que cada uno pueda determinar en qué se ha trabajado el día anterior y poder tener un seguimiento del avance del proyecto.

### **11.2.2. EXCEL**

Antes de empezar con la implementación del proyecto, el responsable técnico del proyecto junto con el programador realiza una planificación de cada una de las tareas a lo largo del tiempo.

En la hoja de cálculo se dibuja un Gantt, en el que se desglosan las tareas y la duración de cada una de ellas, así como la fecha de inicio y la fecha de fin. De esta manera se puede detectar cualquier retraso en la fase de desarrollo.

### **11.2.3. TRELLO**

En Trello se detallan cada una de las tareas del proyecto y nos ayuda, con la vista Kanban que nos ofrece, poder tener una vista general del estado del proyecto.

En cada tarjeta a parte de asignar el encargado de realizarla, se pueden realizar apuntes de la manera en la que se debe realizar, así como de algunos errores detectados en la verificación de cada una de ellas.

### **11.3. DESARROLLO DEL PROYECTO**

La manera en la que se gestiona un proyecto en Develoop varia dependiendo del software a desarrollar, del tipo de cliente y de la dimensión del proyecto. Pero como norma general hay una manera de proceder genérica para todos los proyectos. En la que se tiene como plantilla una metodología en cascada, aunque cada vez más, se están añadiendo algunas técnicas ágiles como pueden ser entregas parciales o incluso adaptación de algunas peticiones realizadas por el cliente. Las distintas fases que detallaremos en los siguientes puntos son:

1. Contacto comercial.
2. Presupuesto.
3. Contrato.
4. Análisis funcional y diseños.
5. Desarrollo del proyecto.
6. Entrega.
7. Mantenimiento.

### **11.3.1. CONTACTO COMERCIAL**

Este es el primer paso de todos, la captación del *lead* o primer contacto con el cliente la realizan los comerciales, que según el tipo de proyecto se encarga uno de este contacto u otro, repartiéndose estos según experiencia en tipos de proyecto.

Este es el punto en el que se recibe una solicitud por parte del cliente, que requiere de un software que resuelva una problemática. Es por eso que el comercial se encarga de una primera evaluación del problema, entender las necesidades y dar una primera solución o soluciones viables que podrían adaptarse a la necesidad del cliente. O por lo contrario descartar la petición por incompatibilidad con la empresa u otros proyectos.

### **11.3.2. PRESUPUESTO**

El comercial se reúne con el director técnico y con el director para tratar el lead y conjuntamente buscar una solución tecnológica viable y adecuada a las necesidades presentadas por parte del cliente.

Para esto el director técnico estima en horas de trabajo, cada una de las tareas de programación que junto con las horas de gestión que determina el director se realiza una primera valoración. Esta primera valoración se plasma sobre un documento, el presupuesto, en el que se estima el coste del proyecto según las partidas determinadas por el equipo.

Una vez finalizado el presupuesto, se presenta al cliente.

Este proceso se puede ir iterando, o no, según las peticiones del cliente, ya sea ajustar un poco más los costes del proyecto o añadir nuevas funcionalidades no previstas con anterioridad.

### **11.3.3. CONTRATO**

Una vez el cliente se decide a la contratación de los servicios de Develoop Software, se plasma el acuerdo en un contrato, en el que entre otras cosas, se estipula el tiempo de entrega del software.

Este contrato se firma por ambas partes, por los representantes legales de la empresa, dando a entender que ambos están de acuerdo con los términos y condiciones que aparecen en él.

### **11.3.4. ANALISIS FUNCIONAL Y DISEÑOS**

Con la aprobación del presupuesto, se realiza una primera reunión, presencial a poder ser, para captar todas y cada una de las funcionalidades de las que hay que dotar el *software*.

Con todo el *feedback* extraído de la reunión se realiza un análisis funcional lo más detallado posible de tal manera que tanto programación como el cliente, al que se le va a enviar también, entiendan a la perfección lo que se debe realizar y como va a quedar.

En paralelo a la realización del análisis funcional, se realizan los *mock-ups* y esbozos de diseños, cuya intención es añadirlos en el análisis funcional para que el cliente tenga una primera impresión de como van a quedar a nivel visual es software.

### **11.3.5. DESARROLLO DEL PROYECTO**

Una vez entregados los diseños y el análisis funcional al equipo de programación el proyecto, lo primero que realizan es una estimación temporal detallada de cada una de las tareas y el tiempo dedicado. Una vez con esta estimación temporal y la segregación del proyecto en distintas tareas, se monta el Trello con todas estas tareas.

El equipo técnico es el encargado de priorizar las tareas, y a partir de aquí se empieza con el desarrollo del proyecto. No hay reuniones programadas a lo largo del proyecto, pero si surge cualquier duda o complicación siempre se mira de programar una reunión entre el equipo de gestión y el de programación, y en caso necesario, el equipo de gestión se pondría en contacto con el cliente para aclarar las dudas.

Como hemos comentado, cada vez más, se opta por realizar entregas parciales al cliente para que este dé el visto bueno a las funcionalidades presentadas, y que esté al día del estado del desarrollo del proyecto.

Cabe resaltar que todo el desarrollo se realiza en un entorno controlado, es decir un entorno de pruebas que posteriormente, con la verificación del cliente y las pruebas de QA correspondientes, este se subirá al entorno de producción.

#### **11.3.6. ENTREGA**

Una vez finalizado el desarrollo y con el visto bueno del cliente se realiza el *deployment* del proyecto en el entorno de producción, en el caso de aplicación web, con el entregable e instalación en el entorno de producción en caso de aplicación desktop o publicación (Si el cliente lo desea) o entrega del ejecutable en el caso de aplicación móvil.

Cuando el *software* se encuentra en el entorno de producción, se realizan una serie de comprobaciones técnicas para revisar el buen funcionamiento del software.

#### **11.3.7. MANTENIMIENTO**

El software entregado por Develoop Software tiene una garantía de 1 año, lo que quiere decir que si surgiera cualquier incidencia o funcionalidad no deseada o un pequeño cambio (previa valoración) a realizar en el proyecto entregado Develoop se hace cargo a coste cero.

Por otra parte, si el *hosting* contratado por el cliente, Develoop se encarga de todas las actualizaciones necesarias para el buen funcionamiento del programa, así como cualquier modificación o incidencia en el servidor en el que está alojado.

## 11.4. ODOO PROYECTOS

La nueva herramienta a incorporar en la empresa es Odoo [23], un ERP *open source* que no únicamente nos va a aportar todas las funcionalidades propias de un ERP sino que también se le ha incorporado un módulo para la gestión de los proyectos, llamado Proyectos. En este punto se va a definir todo lo que esta herramienta aporta para, posteriormente, poder llegar a un punto en que la metodología seleccionada se adapte perfectamente a los beneficios que aporta Proyectos.

### 11.4.1. ORGANIZACIÓN DEL PROYECTO

Todo proyecto se divide en tareas más pequeñas para poder estimar el coste en horas de la realización de cada una de ellas y también para poderlas tratar de manera individual.

Proyectos ofrece una interfaz siguiendo el modelo *Kanban* el cual proporciona, de manera rápida, un gran conocimiento sobre el estado del proyecto. Donde cada tarea está asignada a al estado (columna) en el cual se encuentra en ese momento. Como podemos ver en la Ilustración 7 cada proyecto tiene su propio tablero con las columnas definidas para cada proyecto según las necesidades de este.

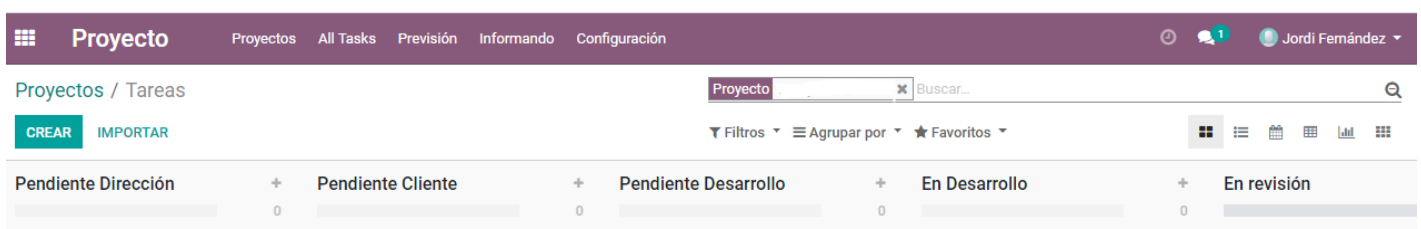


Ilustración 7. Vista Kanban en Odoo. Elaboración propia.



Como hemos comentado cada columna estará compuesta de tareas en la que se podrán añadir distinta información (Ilustración 8):

- Quien realiza la tarea, y seguidores de la tarea.
- Fecha límite para acabarla.
- Subtareas relacionadas con esa tarea.
- Imputar las horas dedicadas a esa tarea.
- Etiquetas, por si se requiere clasificar la información de una manera interna.
- Información general, descripción de la tarea.
- Adjuntos para ayudar a la realización de la tarea.
- Planificar actividades.
- Etapa en la que se encuentra.

Proyectos / Tareas / Pruebas e Instalaciones

EDITAR CREAR Acción 1 / 5 < >

ASIGNARME A MÍ COMPARTIR PENDIENTE DIRECCIÓN PENDIENTE CLIENTE PENDIENTE DESARROLLO EN DESARROLLO EN REVISIÓN EN PRODUCCIÓN

0 Sub-tarea \$ Sales Order Activo

## ☆ Pruebas e Instalaciones

Proyecto Asignada a Elemento del pedido de venta Fecha límite Etiquetas

Descripción Partes de horas Información extra

Enviar mensaje Poner una nota Planificar actividad 0 Seguir 2

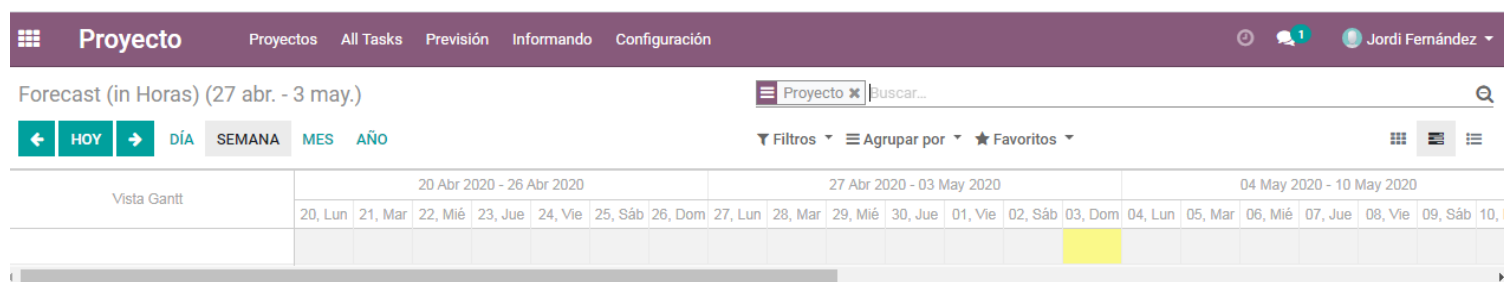
**Ilustración 8. Ficha de tarea. Elaboración propia**

De esta manera toda la información relacionada con el proyecto y sus tareas, estará almacenada en el tablero de cada proyecto, así como la contabilización de las horas dedicada a cada tarea, que una vez llegado el momento estos datos puede ser reveladores de cara a realizar una fase de *review* del proyecto.

### 11.4.2. PLANIFICACIÓN

La realización de los planes y por lo tanto el control de todo lo detallado en él, es uno de los puntos más importantes en la gestión de cualquier proyecto, todo aquello planeado debe seguirse al pie de la letra preocupándose de no salirse del guión en ningún momento.

Es por eso que el módulo ofrece una funcionalidad la cual permite realizar algo parecido a un diagrama de Gantt en el que se visualiza un desglose de todas las tareas del proyecto y las horas planeadas que se van a dedicar a cada una de ellas así como el inicio de la tarea y el fin previsto o si incluso hay alguna dependencia entre tareas o si hay en algún momento en que dos tareas o más se van a realizar en paralelo, si la disponibilidad de los recursos asignados lo permite. (Ilustración 9)



**Ilustración 9. Vista Gantt en Odoo. Elaboración propia.**

### 11.4.3. CALENDARIO

El calendario que incorpora el módulo permite conocer las fechas denominadas “críticas” que permite conocer cuando se acaba una tarea o aquellas

actividades planificadas con anterioridad independientemente del proyecto ya que se muestran todas las fechas sin discriminación alguna. Pero si se requiere se pueden filtrar todos los hitos según proyecto, empleado u otros parámetros.

#### **11.4.4. SEGUIMIENTO DEL PROYECTO**

Para evolucionar y aprender es necesario revisar todos y cada uno de los proyectos y analizar que se ha hecho bien y que se ha hecho mal comparando todo lo previsto y si realmente se ha desarrollado según se había concretado.

Es por eso que Proyectos ofrece la oportunidad de crear gráficas de todo tipo con el fin de facilitar la evolución del proyecto a lo largo de su duración.

De estas gráficas podemos sacar conclusiones como pueden ser:

- Cargas de trabajos asignadas a cada trabajador.
- Horas dedicadas a las tareas, con el fin de compararlas con las estimadas.
- Costes de cada uno de los proyectos.

No únicamente se pueden sacar estos datos, sino que se pueden realizar gráficas con cualquiera de los datos que se introducen en el proyecto siempre y cuando se realicen de manera correcta cada una de las búsquedas, para que la gráfica aporte datos relevantes.

## 12. DECISIÓN FINAL

Una vez analizadas todas las variables que competen una selección adecuada de la metodología a implementar en la empresa, ha llegado el momento de tomar una decisión final en la que escogemos una de las metodologías explicadas con anterioridad y la adaptaremos según las necesidades de Develoop. Uno de los puntos clave a la hora de escoger la metodología es que esta resuelva o al menos reduzca los problemas con los que Develoop se ha ido encontrando a lo largo de los desarrollos de los proyectos, y que hemos definido con anterioridad.

Una de las cosas claras de las que nos hemos dado cuenta a lo largo de este proceso de análisis es que la metodología que seleccionemos será una metodología ágil. Las metodologías ágiles nos proporcionan un control sobre todo aquello que realizamos que vendrá muy bien a la empresa ya que en algunos proyectos vemos que las horas que se planificaron para el desarrollo del proyecto, al cabo del proyecto acaban siendo muchas más y por lo tanto desemboca en una pérdida de dinero. Es por eso que algunos proyectos no acaban de ser viables económicamente.

También destacamos de las metodologías ágiles la importancia que se le da a la figura del cliente. Como empresa que se dedica al *SaaS (Software as a service)* una de nuestras prioridades es que el software se adapte a todas las necesidades del cliente, y la mayor satisfacción de Develoop es que el cliente esté totalmente satisfecho con el *software* desarrollado, por lo que si el programa requiere de algunas modificaciones no contempladas en el inicio, solemos valorar positivamente que estas entren dentro del proyecto con tal de contentar al cliente, siempre y cuando creamos que no suponen un cambio drástico en el desarrollo de este, ya que normalmente Develoop suele correr con los gastos que estos cambios comportan.

Por el contrario, descartamos el uso de las metodologías tradicionales: PMBOK y PRINCE2, por las mismas razones por las cuales nos decantamos por las metodologías ágiles, por el control del tiempo y seguimiento del proyecto que las metodologías ágiles implican. Además, las metodologías en cascada son más reacias a modificaciones o cambios, que deben estar debidamente

definidos desde el inicio del proyecto y no pueden variar ya en la fase del implementación o desarrollo del proyecto.

Por lo que según lo comentado con anterioridad creemos que el tipo de metodología que podemos llegar a seguir e implantar es **SCRUM**.

Todo y que *Kanban* sea una metodología ágil y estemos de acuerdo en algunos puntos, como puede ser el “*Stop starting, start finishing*” y la visibilidad del estado del proyecto sencilla y eficaz que ofrece, uno de los principales puntos importantes a tener en cuenta por parte de Develoop es la fecha de entrega final de proyecto que se estipula en el contrato. En *Kanban*, la calidad tiene una alta importancia, pero esto es en detrimento al tiempo estipulado en ello, ya que no existen *deadlines*, por lo que se nos antoja muy difícil la planificación temporal y por consecuencia el seguimiento temporal y estricto de los tiempos concretados en dicha planificación y contrato.

Referente a la metodología AUP, creemos que tenía potencial suficiente para poder ser la elegida a medida que íbamos sabiendo más de ella y que cumplía con creces lo que esperamos de la metodología, pero creemos que la metodología SCRUM puede aportar mucho más que AUP. Además, viendo poco grado de definición que tiene AUP consideramos que en SCRUM tenemos todo el proceso de proyecto más definido.

Requisitos	SCRUM	PRINCE2	PMBOK	AUP	KANBAN
Deadlines	Medio	Alto	Alto	Medio	Bajo
Definición de roles	Alto	Alto	Alto	Alto	Alto
Control de tiempo	Alto	Alto	Alto	Alto	Alto
Flexibilidad	Alto	Bajo	Bajo	Alto	Alto
Grado de definición	Alto	Alto	Alto	Bajo	Alto
Proyectos Odoo	Medio	Alto	Alto	Medio	Alto

**Tabla 8. Comparativa de metodologías. Elaboración propia.**

Las reuniones que se realizan a diario (*dailys*) con el equipo de desarrollo, nos puede ayudar a determinar cualquier pequeño retraso del proyecto, y ponerle remedio cuanto antes para que la fecha de entrega final no se vea comprometida.

Las entregas parciales que se realizan al final de cada sprint, nos ayudan a:

1. Mantener el contacto con el cliente, y su implicación en el proyecto.
2. Tener un feedback al final de cada sprint, y no al final de todo el proyecto.

No solo eso, sino que marcar estos hitos intermedios en el proyecto nos mantiene a todo el equipo perfectamente sincronizado y nos marquen un avance del proyecto constante y regular para llegar a tiempo a la entrega final.

Si es verdad que en SCRUM en algunos proyectos, no se cuenta con una fecha final, pero en nuestro caso esta fecha es necesaria ya que es una de las condiciones que Develoop se compromete a cumplir al firmar el contrato. Y es por eso que deberemos buscar soluciones a lo largo de la implementación que nos ayuden a mantener esta fecha final, y una planificación adecuada, en la que se tenga en cuenta las posibles modificaciones al final del sprint.

Aunque ahora, las reuniones *post sprint* parezcan una pérdida de tiempo debemos pensar en el futuro, ya que de estas se pueden aprender cosas a mejorar de cara a futuros proyectos o incluso correcciones a realizar en el propio desarrollo de un proyecto activo y ayude a mejorar el rendimiento y optimizar el tiempo.

Además, creemos que la nueva herramienta incorporada en la empresa y la vista *Kanban* que ofrece, se puede adaptar perfectamente a esta metodología.

Es por esta seguridad y sensación del control del proyecto y el avance de este que creemos que SCRUM puede ayudarnos a paliar los problemas presentados con anterioridad, y sacar *software* de mayor calidad.

## **13. IMPLEMENTACIÓN DE LA METODOLOGIA**

Este es el punto más importante de todo el TFG, ya que es la piedra angular del proyecto, el punto donde todo lo analizado, detallado y explicado convergen en un objetivo común: la implementación de la metodología.

Aquí se detallarán todos y cada uno de los nuevos procesos que se van a seguir en la gestión de un proyecto y la manera en que se van a llevar acabo así como las herramientas de soporte que se van a utilizar a lo largo de todo el proyecto.

Habrán algunos pasos que se seguirán de la misma manera en la que se están realizando ahora, pero lo que realmente se debe detallar y concretar a la perfección mirando de dejarlo todo claro sin dar pie a malentendidos.

Vamos a numerar los las distintas fases que vamos a abarcar en la implementación de la metodología, que serán las que ya mencionamos en su momento:

1. Contacto comercial.
2. Presupuesto.
3. Contrato.
4. Análisis funcional y diseños.
5. Desarrollo del proyecto.
6. Entrega.
7. Mantenimiento.

### **13.1. CONCEPTOS PREVIOS**

En este punto previo al detalle del proceso es conveniente resaltar el concepto “Triángulo de hierro”. Este triángulo muestra la relación entre las tres variables básicas de un proyecto y la gestión a través del sistema de calidad, que se

encargará de buscar el equilibrio entre ellos, buscando siempre que sea un triángulo equilátero.



**Ilustración 10. Triángulo de hierro.**

**Recuperado de:** <http://gestiondeproyectos-master.com/el-triangulo-de-hierro-en-gestion-de-proyectos/>

La variable de alcance se trata del producto final a entregar, que de forma ideal se definen al inicio del proyecto.

El plazo o tiempo es la fecha de entrega del alcance del proyecto, aunque no necesariamente debe ser la entrega final, sino que pueden ser los plazos relacionados con las entregas parciales.

El coste es el presupuesto dedicado al proyecto, el precio de la realización completa del producto final.

Y como bien hemos dicho, la calidad se encarga de la gestión de estas tres variables, una mala calidad influirá de manera negativa en las tres variables.  
[25]

Una de las características de las metodologías ágiles es la flexibilidad, y es por eso que hay que tener presente en todo momento este triángulo porque cualquier modificación en los requisitos (alcance) implicará una actualización de las otras dos variables.

En todo proyecto Scrum, existen dos roles necesarios y claramente diferenciados: *Product Owner* y *Scrum Master*. El rol de *Product Owner* lo desempeñará el cliente, y el rol de *Scrum Master* lo desempeñará el *Project Manager* asignado al proyecto.



Teniendo presente estos conceptos, procederemos al detalle de las distintas fases mencionadas con anterioridad.

### **13.2. CONTACTO COMERCIAL**

Aprovechando Odoo al máximo, la organización de los *leads* se va a realizar mediante el módulo del CRM. A medida que vayan entrando los leads, ya sea directamente al CRM, por correo electrónico o por teléfono. Todos ellos deberán estar organizados en el módulo de CRM, no tan solo para coordinar la llamada al *lead* sino también para tener un historial, y en un futuro poder rescatar información acerca del contacto o consultar cualquier información de interés.

### **13.3. PRESUPUESTO**

Desde Odoo se pueden realizar presupuestos, que en un futuro se podrán aprovechar todas las partidas detalladas, que cuando un presupuesto es aprobado, pasaran directamente al módulo de Proyectos como tareas del proyecto (de esto hablaremos en siguientes puntos).

El proceso, va a ser el mismo que el detallado en el punto 11:

Se consultara con el director técnico una aproximación de las horas destinadas a cada una de las partidas y el Director añadirá las horas de gestión que prevé que se dediquen a lo largo de todo el proyecto.

Todas estas horas se reflejaran en el presupuesto, y se entregara al cliente para que acepte o rechace el presupuesto.

A partir del presupuesto inicial, se realizarán iteraciones, si la situación lo requiere: el cliente pide que se ajuste el presupuesto o se eliminan funcionalidades.

Cuanto más desglosadas y detalladas estén las partidas en el presupuesto, menos faena se tendrá después para montar el entorno de seguimiento del proyecto en Odoo.

### **13.4. CONTRATO**

En este documento se plasman todos los términos y condiciones del servicio ofrecido por parte de Develoop así como los plazos de facturación del proyecto y el plazo de entrega de este.

### **13.5. ANALISIS FUNCIONAL Y DISEÑOS**

Para poder rellenar el *Product Backlog* con los requerimientos del proyecto, se realizará una reunión con el cliente para definir todos y cada uno de los requisitos los cuales el *software* tiene que estar dotado.

Para ello se realizará una análisis funcional que servirá como referencia a lo largo del proyecto para poder consultar, y es de este donde se extraerán los requisitos del proyecto que irán a parar al *Backlog* general.

En paralelo a la realización de análisis funcional, se realizan *mock-ups* que se van validando con el cliente, para que de cara a la presentación del análisis funcional al cliente, incluir como mínimo en *mock-ups* y en caso ideal tener los diseños a la par que el análisis funcional, para que el equipo de programación disponga de todos los materiales para empezar con el desarrollo del proyecto

### **13.6. DESARROLLO DEL PROYECTO**

Este es el punto más importante de todos los que hemos mencionado es por eso que, con el fin de detallarlo mejor, vamos a diferenciar claramente las los distintos procesos que componen el desarrollo del proyecto.

Además, es en este punto donde hay que tener en cuenta el “Triángulo del hierro” ya que cualquier petición extra del cliente a añadir en el *Product Backlog*

debe ser valorada previamente por el *Scrum Master* para determinar cuál de la(s) variable(s) debe redefinirse. En el caso de aumentar el alcance del proyecto habrá que modificar las variables de tiempo y coste, siendo lo más lógico aumentar ambas variables: dando más tiempo para la realización del proyecto y por consecuencia aumentando el presupuesto.

En el caso concretado en el párrafo de arriba, habría que realizar un pequeño presupuesto incluyendo los nuevos requisitos, la nueva fecha de entrega y el coste de cada uno de las partidas.

Todos los datos y credenciales asociados al proyecto se irán apuntando en el espacio reservado que tiene Develoop destinado para ello, al que todos los integrantes del equipo de proyecto tengan acceso.

Con el presupuesto aceptado desde Odoo, se nos creará automáticamente el entorno de seguimiento del proyecto, por lo que todas las partidas detalladas en el presupuesto, se nos darán de alta como tareas en el tablero del proyecto.

Con el proyecto ya creado en Odoo, se procederá a realizar una primera planificación temporal de todo el proyecto. Hay que tener en cuenta, que para las metodologías ágiles, y en este caso SCRUM, las planificaciones, así como los requisitos, son variables, por lo que dependiendo de los nuevos requisitos añadidos, la planificación variará. Pero esta planificación inicial, siempre nos pueden servir de punto de partida.

Previamente, al montaje del tablero de seguimiento del proyecto, se definirá la longitud de cada *sprint* dependiendo del alcance del proyecto y el tiempo estimado para la entrega final.

Es en este punto cuando se deberá montar el *Product Backlog* del proyecto, es por hecho que aprovecharemos las partidas del presupuesto para convertirlas en las épicas del proyecto.

En el tablero por defecto definiremos los estados de las tareas, si algún proyecto requiere de más estados se añadirán. Estos estados serán:

- **Épicas**. Aquí se colocaran las partidas aprobadas del presupuesto. Se partirá de estas macrotareas, para la creación de las otras. Para ello se crearán de

cada una de las épicas. Para ello Odoo ofrece la opción de crear de una tarea, subtareas relacionadas con ellas.

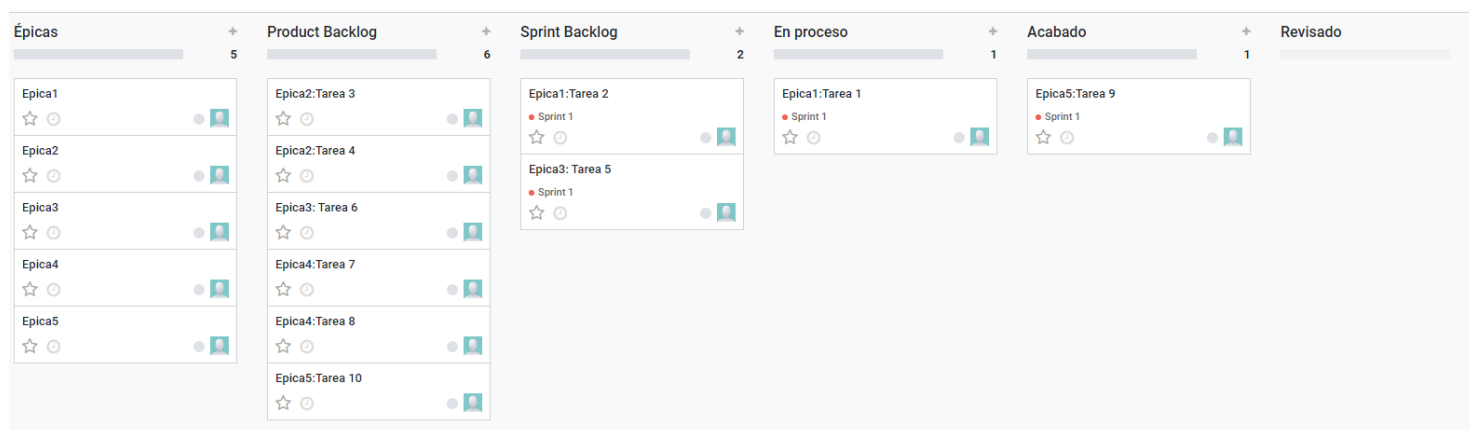
- **Product Backlog.** En esta columna se colocaran todas las subtareas que conjuntamente determinan el alcance del proyecto. A cada tarea, habrá que asignarle las horas planificadas para la finalización de la tarea.

- **Sprint Backlog.** Al comienzo de cada *sprint* se colocaran en esta columna las tareas a realizar, que se cogerán del *Product Backlog*. Si hay tareas que no se han finalizado en el *sprint* anterior, pasarán a formar parte del siguiente *sprint*.

- **En proceso.** A medida que se vayan empezando tareas, estas irán pasando a esta columna.

- **Acabado.** Cuando una tarea se haya finalizado, pasará al estado “acabado”. Las tareas que estén en esta columna, deberán ser revisadas por el *Scrum Master* del proyecto, y él mismo es el que pasará las tareas a revisado, en caso de estar bien desarrollada o a “en proceso” en caso de realizar algunas correcciones que se comentarán en la tarjeta aportando la mayor información posible.

- **Revisado.** Columna en la que aparecerán las tareas consideradas acabadas por el *Scrum Master*.



**Ilustración 11. Organización proyecto. Elaboración propia.**

En cada tarea se deberá detallar las horas planificadas en la realización de cada una de ellas, de esta manera aparte de tener las horas contabilizadas en cada tarea por cada integrante del equipo de proyecto, podremos ver el margen de tiempo con el que se cuenta para la realización de la tarea, o en caso contrario las horas de más dedicadas en la tarea.

Adicionalmente, como hemos comentado todos los integrantes al finalizar el día, deben contabilizar las horas realizadas en cada una de las tareas, pudiendo añadir una descripción de lo realizado. Esto ayudará el *Scrum Master* a tener una idea en lo que se trabajó ayer, y a entrar más en detalle en el estado del proyecto, pudiendo tomar decisiones justificadas en caso oportuno.

Una vez montado todo el tablero, se podrá proceder con el desarrollo del proyecto empezando por el *Sprint Planning* que es la reunión que se realiza al inicio de cada *sprint* para determinar que tareas se van a realizar en cada uno. Para ello se marcarán mediante una etiqueta a que sprint pertenecerá cada tarea. Esto nos permitirá el filtro por etiquetas, para mostrar únicamente aquellas tareas propias de cada sprint, como para sacar posteriormente datos necesarios para la elaboración de gráficos.

Durante el desarrollo de todo el proyecto, se mantendrán diariamente reuniones al inicio del día en las que participará todo el equipo de proyecto en el que cada integrante deberá responder a las tres preguntas detalladas en puntos anteriores:

- 1) Que tareas completé ayer
- 2) Que tareas voy a realizar hoy
- 3) Impedimentos que no me han permitido avanzar.

Estas *daily*s ayudarán también a detectar posibles retrasos o cualquier otro problema relacionado con el avance del proyecto.

Una vez finalizado el *sprint*, se realizarán las dos reuniones claves para revisar como ha ido el *sprint*:

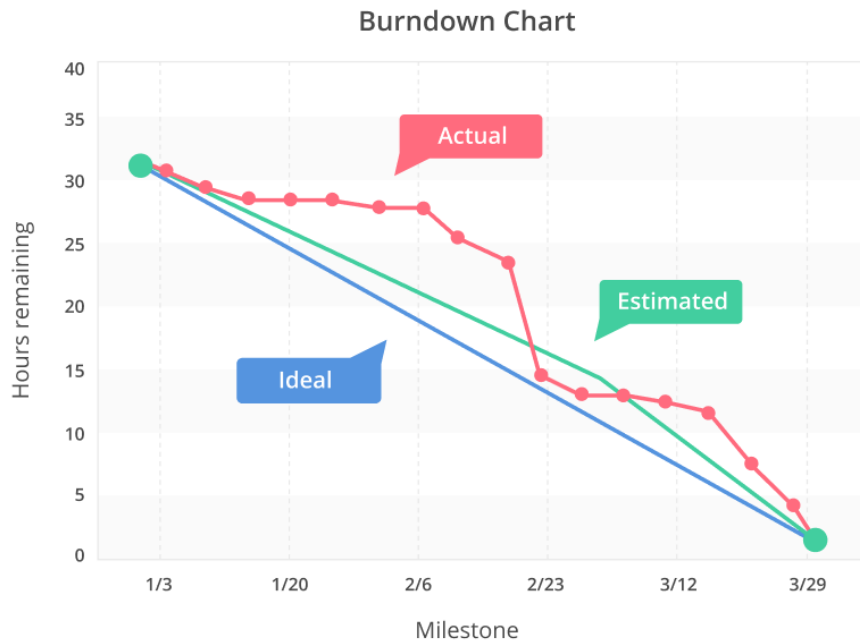
- ***Sprint review***. En esta reunión con el Product Owner, y el equipo de proyecto de Develop se realizará una demostración del producto desarrollado

al final de este *sprint*, en la que el Product Owner mostrará sus impresiones y feedback a tener en cuenta en el desarrollo del proyecto. También se analizará el *Product Backlog* faltante para poderlo organizar, que junto con el feedback captado será útil para el *Sprint Planning* siguiente, en este punto si se considera que algunas de las tareas no están correctamente finalizadas pasaran directamente al siguiente *Sprint Backlog*. Además, se realizará una revisión del presupuesto, tiempo y alcance del proyecto a tener en cuenta en los próximos *sprints*.

- ***Sprint retrospective***. Esta reunión se realizará entre los miembros de Develoop Software para poder comentar posibles mejoras en el rendimiento del equipo, detección de problemas y posibles soluciones. Esta reunión se realiza con el fin de mejorar en los próximos *sprints* y en futuros proyectos, es una reflexión del desenlace del proyecto.

En la finalización del sprint puede ser de gran ayuda la creación de informes que ayuden a entender cómo ha ido el sprint, es por eso que se añadirán a la ficha del proyecto los siguientes informes:

**Burndown Chart**. El burndown chart es una gráfica que nos muestra una evolución de las horas planificadas a realizar en este sprint en relación al tiempo. Se puede ver cómo se van consumiendo las horas estimadas para la realización de las tareas de este *sprint*. De este informe se pueden extraer si ha habido algunas tareas que han demorado más tiempo de lo previsto, o por si lo contrario se ha realizado según lo estimado.



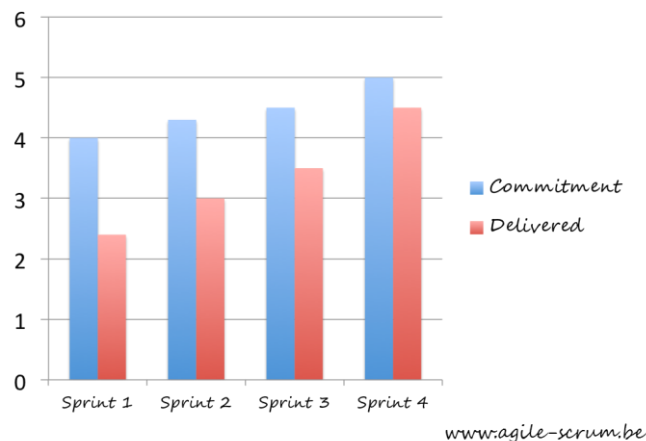
**Ilustración 12. Ejemplo Burndown Chart.**

**Recuperado de:** <https://backlog.com/blog/burndown-charts-help-keep-projects-target-heres-get-started/>

Esta gráfica la obtendremos teniendo en cuenta todas las tareas, que vayan pasando al estado “Revisado”, a las cuales se les guarda el día en el que la tarea pasa de estado. [24]

- **Velocity chart.** Esta es una gráfica comparativa entre *sprints*, en la que por cada sprint, se comparan las horas planeadas a realizar en el sprint con las horas realmente realizadas. Esta gráfica se va actualizando a medida que se van realizando *sprints*. La idea es ir ajustando las horas a realizar en cada sprint para ir determinando con más precisión que tareas realizar en cada sprint, de tal manera que se realicen el máximo de tareas posibles sin pasarse. Al final del proyecto, tendremos una comparativa de la sucesión de los *sprints*, en la que en principio las horas estimadas y las reales deberían irse acercándose lo máximo posible a medida que el proyecto va avanzando.

## VELOCITY CHART



**Ilustración 13. Ejemplo Velocity Chart.**

**Recuperado de:** <https://www.agile-scrum.be/whats-great-scrum-methodology/velocity-chart/>

Estos informes no se pueden generar de forma automática en Odoo, pero lo que sí que podemos extraer los datos necesarios para la realización de estas gráficas. Se extraerán de Odoo en formato Excel y posteriormente se procedería a plasmar estos datos en un gráfico.

### 13.7. ENTREGA

Una vez finalizado el producto, se realiza la entrega del desarrollo al cliente. Como se han ido realizando demostraciones y entregas parciales, a nivel funcional no existirán muchas desviaciones. Para ello se validará junto con el *Product Owner* si el producto cumple con su cometido y se procederá a su despliegue en el servidor de producción, y en el caso de Apps, con la publicación en las Stores (App Store y Google Play) correspondientes.

En algunos casos a las Apps se les van a dar un uso interno, por eso no se requiere su publicación en tiendas, sino que en el caso de Android se distribuye el ejecutable y el caso de IOS se sube a TestFlight lo que nos permite controlar quien tiene acceso a la aplicación móvil.



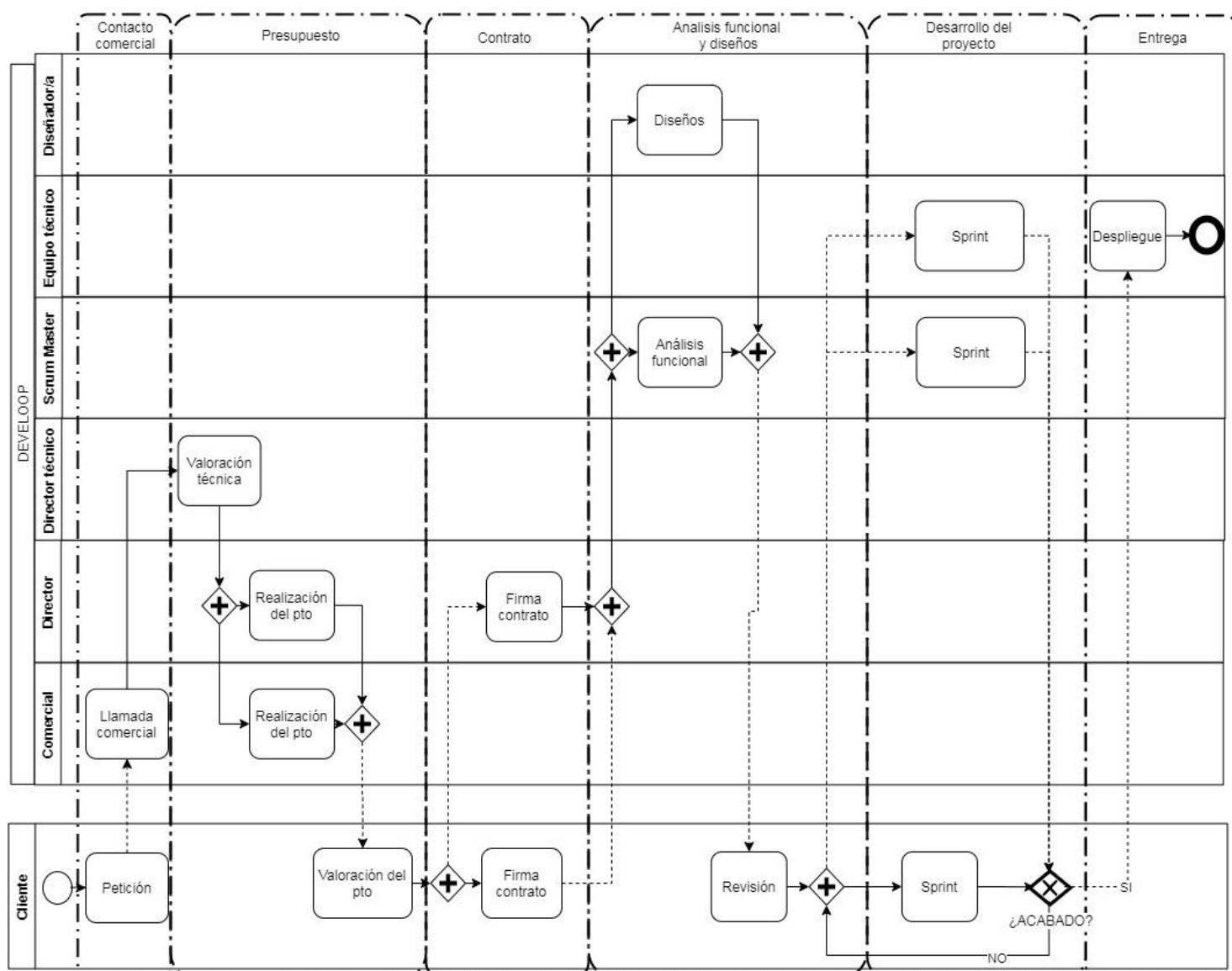


Ilustración 14. BPMN nueva metodología. Elaboración propia.

### 13.8. MANTENIMIENTO

Esta etapa únicamente se desarrollará si el cliente contrata este servicio. Este servicio incluye el *hosting*, monitorización y actualización de versiones de la tecnología del producto realizado.

Contratando este servicio Develoop Software se compromete a la resolución de cualquier incidencia detectada por parte del cliente o mediante la monitorización del servidor.

## **14. CONCLUSIÓN FINAL**

### **14.1. DESVIACIONES**

No existe ningún proyecto en que las horas planeadas a realizar realmente concuerden con las horas reales dedicadas al proyecto.

Este proyecto no ha sido una excepción, por lo que en este apartado expondremos las desviaciones temporales agrupándolas por grupos de tareas que definimos en la fase de GEP. Aunque, el total de horas destinadas al proyecto no ha sido tan dispar a las estimadas de un inicio, ya que se aplicaron márgenes para afrontar posibles problemas durante el proyecto.

Para ello, todas las horas dedicadas a este proyecto se han ido contabilizando en Odoo, en cada una de las tareas concretadas desde un inicio:

#### GESTIÓN DEL PROYECTO

Este grupo de tareas, comprendían la fase de planificación del proyecto y contextualización, así como las reuniones de seguimiento con el director del TFG.

Debido a la situación extraordinaria que hemos estado viviendo, el número de reuniones han aumentado, siempre para favorecer el buen control y seguimiento del proyecto. Por lo que las horas contabilizadas para esta fase son de 87.

La tarea asignada a las reuniones ha aumentado en el número de horas, debido a que se han realizado una reunión por semana, y no cada dos como se tenía planteado.

#### ANÁLISIS

A lo largo del tiempo esta es una fase que se ha extendido, pero no por un gran incremento de horas sino porque durante un periodo de tiempo, por problemas personales, no me he podido dedicar el tiempo deseado en el desarrollo de

estas tareas. Sin embargo, la diferencia de horas entre las planificadas y la reales no es muy significativa ya que se han registrado un total de 203.25 horas, respecto a las 185 estimadas.

### IMPLEMENTACIÓN

Para cumplir con la planificación temporal, a este grupo de tareas se han dedicado más horas diarias con tal de recuperar el tiempo perdido en el anterior grupo. Es por eso que en la planificación temporal real, este grupo de tareas se han visto reducido en el tiempo, aunque se hayan dedicado algunas horas de más respecto a las planificadas. En total se han dedicado 201.5 horas.

### FASE FINAL

Como esta fase aun no se ha realizado, en la que se incluye la realización de la presentación en concordancia al trabajo, consideraremos que las 20 horas estimadas desde un inicio son suficientes.

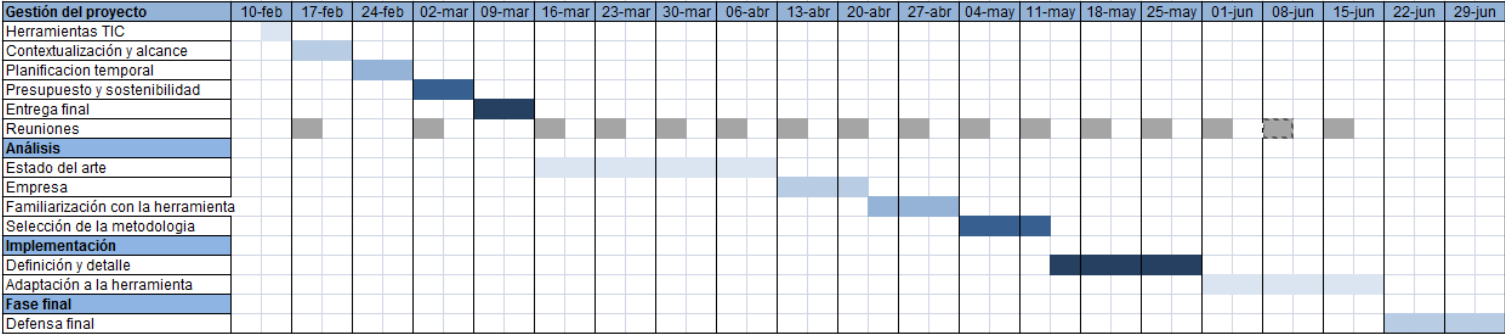


Tabla 9. Diagrama de Gantt real. Elaboración propia.

Teniendo en cuenta estas desviaciones, y que la defensa final podría realizarse el 29 de Junio el diagrama de Gantt real del proyecto ha sido el siguiente:

Las desviaciones no tan solo han tenido impacto en las horas, sino que también en los costes del proyecto, al fin y al cabo, el tiempo es dinero.

La desviación total en horas del proyecto ha sido de: 11.75h, teniendo en cuenta que las 11 horas de más surgidas por las reuniones de seguimiento han sido realizadas conjuntamente con un *Project Manager Senior*. Es por eso que el coste adicional del proyecto ha sido de:

Analista/*Project Manager* Junior –  $11.75 * 9.5\text{€/h} = 111.63\text{€}$

*Project Manager* Senior –  $11 * 20.5\text{€/h} = 225.5 \text{ €}$

**Total** =  $111.63\text{€} + 225.5\text{€} = 337.13 \text{ €}$

Debido a la situación anormal, que hemos vivido estos últimos meses los costes genéricos estimados han sido menores de lo esperado ya que en el transcurso de esta situación se ha realizado teletrabajo, por lo que el coste de transporte durante 3 meses y medio ha sido nulo, y los gastos de agua, electricidad y alquiler de oficina, al trabajar en casa también se han visto afectados:

Transporte -  $20\text{€/mes} * 1.5\text{meses} = 30\text{€}$

Oficina –  $1400\text{€/mes} * 1.5\text{meses} = 2100\text{€}$

Agua –  $370\text{€/mes} * 1.5\text{meses} + 25\text{€/mes} * 3.5\text{meses} = 642.5\text{€}$

Electricidad –  $500\text{€/mes} * 1.5\text{meses} + 35\text{€/mes} * 3.5 \text{ meses} = 872.5 \text{ €}$

Se estimaron en total un coste de 12375€ que han acabado siendo un total de 4545€.

Teniendo en cuenta estos costes, el proyecto ha salido un poco más barato de lo esperado gracias a la reducción de estos costes genéricos, se tenían estimados 19950.96 y en total han sido de 10018.7€

## **14.2. CONSIDERACIONES FINALES**

A lo largo del desarrollo de todo este proyecto se han cumplido los objetivos estipulados desde un inicio donde el principal era estandarizar una metodología útil para la empresa. No solo esto sino que se han cumplido las competencias técnicas relacionadas con el proyecto las cuales se seleccionaron en un inicio y se confirmaron el hito intermedio:

- CES 1.2. Se ha propuesto una nueva solución a nivel procedimental teniendo en cuenta el hacer de la empresa así como las tecnologías de las cuales Develoop dispone.

- CES 1.3. En la primera fase de GEP, se evaluaron los riesgos potenciales que podrían surgir en el desarrollo del proyecto y se propusieron planes alternativos para su gestión en caso que ocurrieran.
- CES 2.1. Definir y gestionar los requisitos de un sistema *software*.

- CES 2.2. Se ha diseñado la metodología en base a Develoop y sus procesos y no a la inversa, para ello se ha identificado una serie de requisitos o problemas que se deberían solucionar y en base a estos se ha escogido la metodología más adiente.

- CES 1.9. A lo largo del proyecto se ha mirado de tener en cuenta todas las variables que intervenían en el proyecto así como de las tecnologías a usar.

Durante el proyecto, me he ido dado cuenta que posiblemente habría algunas cosas a mejorar, para aprovechar el máximo la metodología Scrum. Estas quedaban fuera del alcance del proyecto, pero creo que es bueno detectarlas para tenerlas en cuenta de cara al futuro.

Todas estas van relacionadas con una personalización más individualizada de la herramienta usada para la gestión de proyectos:

- Se ha detectado que la realización de las gráficas, pensadas para realizar manualmente, se podrían automatizar con el fin de optimizar el tiempo. Ahora está pensado, para sacar los datos necesarios de Odoo, pero realizar las graficas con terceras herramientas. La creación automática de estas gráficas una vez finalizado el *sprint*, beneficiaría a todo el equipo de gestión de proyectos.

- Otra de las posibles mejoras, seria la automatización de los finales de *sprints*, trasladando todas las tareas no finalizadas al *Sprint Backlog* de la siguiente iteración.

### 14.3. REFLEXIONES

A nivel personal este proyecto me ha ayudado a conocer distintas metodologías las cuales no tenía constancia que existieran, e incluso alguna me ha despertado interés en seguir aprendiendo de ella.

Ahora mismo mi futuro lo quiero enfocar al *Project Management* y este proyecto me ha resultado realmente útil para evaluar los pros y contras de cada una de las metodologías así como los casos ideales en que las metodologías son realmente útiles.

Personalmente, esta metodología creo que ayudará a resolver algunos de los problemas existentes en la actualidad, pero esto no lo sabremos hasta que la hayamos implementado en alguno de los proyectos. Y es en este punto cuando realmente nos daremos cuenta de lo importante que es seguir una metodología en la gestión de cualquier proyecto, más incluso cuando se deben coordinar distintos equipos de trabajo.

## 15. BIBLIOGRAFÍA

- [1] Arriaga, Maria Loreto: Normas ISO 9126 – 25000, 15 mayo 2017 [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <<https://es.slideshare.net/MaLoretoArriagada/normas-iso-9126-25000>>
- [2] Develoop Software, Develoop Software SL [en línea]. [Consulta: 18/02/2020]. Disponible en: <<https://www.develoop.net/>>
- [3] Project Management. A: OBS Business School [en línea]. [Consulta: 18 de febrero 2020]. Disponible en: <<https://obsbusiness.school/es/blog-project-management/salidas-profesionales-pm/que-es-un-project-manager-y-cuales-son-sus-tareas>>
- [4] Real Academia Española, Real Academia Española [en línea]. [Consulta: 20 febrero 2020]. Disponible en: <<https://www.rae.es/>>
- [5] García Rodríguez, Manuel José. Estudio comparativo entre las metodologías ágiles y las metodologías tradicionales para la gestión de proyectos software [en línea]. Trabajo fin de máster, Universidad de Oviedo. Departamento de explotación y prospección de minas, Julio 2015 [Consulta: 20 febrero 2020]. Disponible en: <<http://digibuo.uniovi.es/dspace/bitstream/10651/32457/6/TFMMIJGarciaRodriguezRUO.pdf>>
- [6] What Is, Tech Target [en línea]. [Consulta: 24/02/2020] Disponible en: <<https://whatis.techtarget.com/>>
- [7] European Business School. ¿Cuáles son las responsabilidades de un Product Owner?, Abril 2018. [En línea]. [Consulta: 11/04/2020] Disponible en: <<https://www.eude.es/blog/responsabilidades-product-owner/>>
- [8] Canal, Paula. Definición y características del Scrum Master, [En línea] Escuela de Negocios de la Innovación y los Emprendedores, Septiembre 2015. [Consulta: 11/04/2020]. Disponible en: <<https://www.iebschool.com/blog/definicion-y-caracteristicas-del-scrum-master-agile-scrum/>>

- [9] Jose. Qué es Scrum en cinco minutos. Pros y contras. [en línea] El Negocio Digital, Septiembre 2019. [Consulta: 11/04/2020] Disponible en: <<https://elnegociodigital.com/post/que-es-scrum-en-cinco-minutos-pros-y-contras#desventajas-scrum>>
- [10] Sliger, Michele. Agile project Management with Scrum [En línea]. Project Management Institute, Octubre 2011. [Consulta 13/04/2020]. Disponible en: <<https://www.pmi.org/learning/library/agile-project-management-scrum-6269>>
- [11] Prince2. [En línea]Prince2 wiki. [Consulta: 13/04/2020]. Disponible en: <<https://prince2.wiki/es/>>
- [12] Los 5 principales beneficios de utilizar PRINCE2. [En línea] Netmind, Marzo 2014. [Consulta: 13/04/2020] Disponible en: <<https://www.netmind.es/knowledge-center/principales-beneficios-de-utilizar-prince2/>>
- [13] Buehring, Simon. PRINCE2 vs the PMBOK Guide: A comparison. [En línea] Knowledge TRAIN, Abril 2020. [Consulta: 14/04/2020] Disponible en: <<https://www.knowledgetrain.co.uk/project-management/pmi/prince2-and-pmbok-guide-comparison#p2-strengths>>
- [14] Guía PMBOK: definición, estructura y tips de estudio. [En línea] EAE Business School, Mayo 2020. [Consulta: 14/04/2020]. Disponible en: <<https://retos-operaciones-logistica.eae.es/que-es-la-guia-pmbok-y-como-influye-en-la-administracion-de-proyectos/>>
- [15] The Agile Unified Proces (AUP). [En línea] Ambyssoft. [Consulta: 21/04/2020]. Disponible en: <<http://www.ambyssoft.com/unifiedprocess/agileUP.html>>
- [16] Agile Unified Process [En línea] EcuRed. [Consulta: 21/04/2020] Disponible en: <[https://www.ecured.cu/Agile\\_Unified\\_Process\\_>](https://www.ecured.cu/Agile_Unified_Process_>)
- [17] Kanban [En línea]. Data Science Project Management. [Consulta: 22/04/2020]. Disponible en: <<http://www.datascience-pm.com/kanban/>>



- [18] El modelo en cascada: desarrollo secuencial de software [En línea]. Digital Guide Ionos, Marzo 2019. [Consulta: 25/04/2020] Disponible en: <<https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/>>
- [19] En qué consiste el modelo en cascada [En línea]. OpenClassrooms, Febrero 2020 [Consulta: 25/04/2020] Disponible en: <<https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>>
- [20] Pros y contras de la metodología en cascada [En línea]. OBS Business School. [Consulta: 25/04/2020] Disponible en: <<https://obsbusiness.school/es/blog-project-management/metodologia-agile/pros-y-contras-de-la-metodologia-en-cascada>>
- [21] Roselló Villán, Vanessa. Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa. [En línea] IEBS School, Marzo 2019. [Consulta: 25/04/2020] Disponible en: <<https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>>
- [22] Tena, María. ¿Qué es la metodología 'agile'? [En línea]. BBVA, Noviembre 2018 [Consulta: 25/04/2020] Disponible en: <<https://www.bbva.com/es/metodologia-agile-la-revolucion-las-formas-trabajo/>>
- [23] Odoo [En línea] [Consulta: 03/05/2020] Disponible en: <<https://www.odoo.com/>>
- [24] Palavicini, Andres. ¿Para qué sirve el sprint burndown chart? [En línea] LinkedIn [Consulta: 23/05/2020] Disponible en: <<https://www.linkedin.com/pulse/para-qu%C3%A9-sirve-el-sprint-burndown-chart-andres-palavicini/>>
- [25] Canal, Paula. ¿Conoces el triángulo de hierro para tus proyectos? [En línea] IEBS School, Noviembre 2015 [Consulta: 02/06/2020] Disponible en: <<https://www.iebschool.com/blog/triangulo-de-hierro-agile-scrum/>>

## 16. GLOSARIO

**Bug.** Error o defecto en el software o hardware que impide que un programa funcione correctamente.

**Deadline.** Fecha límite.

**Deployment.** Actividades que hacen que un sistema software esté disponible para su uso.

**Épica.** Tarea de alta complejidad, que normalmente se divide en subtareas para tratarla.

**ERP.** *Enterprise Resource Planing*: Sistema de planificación de recursos empresariales

**Feedback.** Comentario, valoración.

**Glassdoor.** Sitio web donde los empleados actuales y anteriores revisan a las empresas.

**Hosting.** Alojamiento web.

**ISO/IEC 9126.** Estándar internacional para la evaluación del software.

**Kanban.** Tarjetas visuales en las que se muestran los elementos, en este caso: tareas.

**Keyword.** Palabra clave.

**Lead.** Usuario que entrega sus datos a una empresa, que pasa a entrar en la base de datos con el que la empresa puede interactuar.

**Mock-up.** Maqueta. Esbozo a escala real.

**Norma ISO/IEC 25000.** Norma que establece criterios para la especificación de requisitos de calidad de productos software.

**Odoo.** Software ERP (*Enterprise Resource Planing*: Sistema de planificación de recursos empresariales)

**Open Source.** Modelo de desarrollo del software basado en la colaboración de la comunidad.

**Partners.** Socio. Relación que se establecen entre dos empresas para obtener un beneficio mutuo.

**Planning.** Planificación.

**QA.** *Quality assurance*, control de calidad.

**RAE.** Real Academia Española

**Review.** Revisión.

**Sprint.** Período breve de tiempo fijo en el que un equipo de Scrum trabaja para completar una cantidad de trabajo establecida.

**TestFlight.** Servicio que ofrece Apple para la instalación y prueba de aplicaciones móviles.

**Timing.** Tiempo estimado.

**WhatIs.** Herramienta de referencia, propiedad de TechTarget, sobre tecnología de la información.